OPTIMIZATION OVER SYMMETRIC CONES UNDER UNCERTAINTY

By

BAHA’ M. ALZALG

A dissertation submitted in partial fulfillment of
the requirements for the degree of

DOCTOR OF PHILOSOPHY

WASHINGTON STATE UNIVERSITY
Department of Mathematics

DECEMBER 2011
To the Faculty of Washington State University:

The members of the Committee appointed to examine the dissertation of BAHA’ M. ALZALG find it satisfactory and recommend that it be accepted.

______________________________
K. A. Ariyawansa, Professor, Chair

______________________________
Robert Mifflin, Professor

______________________________
David S. Watkins, Professor
For all the people
ACKNOWLEDGEMENTS

My greatest appreciation and my most sincere “Thank You!” go to my advisor Professor Ari Ariyawansa for his guidance, advice, and help during the preparation of this dissertation. I am also grateful for his offer for me to work as his research assistant and for giving me the opportunity to write papers and visit several conferences and workshops in North America and overseas.

I wish also to express my appreciation and gratitude to Professor Robert Mifflin and Professor David S. Watkins for taking the time to serve as committee members and for various ways in which they helped me during all stages of doctoral studies.

I want to thank all faculty, staff and graduate students in the Department of Mathematics at Washington State University. I would especially like to thank from the faculty Associate Professor Bala Krishnamoorthy, from the staff Kris Johnson, and from the students Pietro Paparella for their kind help.

Finally, no words can express my gratitude to my parents and my grandmother for love and prayers. I also owe a special gratitude to my brothers and sisters, and some relatives in Jordan for their support and encouragement.
Abstract

by Baha' M. Alzalg, Ph.D.
Washington State University

December 2011

Chair: Professor K. A. Ariyawansa

We introduce and study two-stage stochastic symmetric programs (SSPs) with recourse to handle uncertainty in data defining (deterministic) symmetric programs in which a linear function is minimized over the intersection of an affine set and a symmetric cone. We present a logarithmic barrier decomposition-based interior point algorithm for solving these problems and prove its polynomial complexity. Our convergence analysis proceeds by showing that the log barrier associated with the recourse function of SSPs behaves as a strongly self-concordant barrier and forms a self-concordant family on the first stage solutions. Since our analysis applies to all symmetric cones, this algorithm extends Zhao's results [48] for two-stage stochastic linear programs, and Mehrotra and Özevin's results [25] for two-stage stochastic semidefinite programs (SSDPs). We also present another class of polynomial-time decomposition algorithms for SSPs based on the volumetric barrier. While this extends the work of Ariyawansa and Zhu [10] for SSDPs, our analysis is based on utilizing the advantage of the special algebraic structure associated with the symmetric cone not utilized in [10]. As a consequence, we are able to significantly simplify the proofs of central results. We then describe four applications leading to the SSP problem where, in particular, the underlying symmetric cones are second-order cones and rotated quadratic cones.
Contents

Acknowledgement iv

Abstract v

Chapter 1 1

1 Introduction and Background 1

1.1 Introduction 1

1.2 What is a symmetric cone? 7

1.3 Symmetric cones and Euclidean Jordan algebras 11

2 Stochastic Symmetric Optimization Problems 28

2.1 The stochastic symmetric optimization problem 28

2.1.1 Definition of an SSP in primal standard form 29

2.1.2 Definition of an SSP in dual standard form 30

2.2 Problems that can be cast as SSPs 31

3 A Class of Polynomial Logarithmic Barrier Decomposition Algorithms for Stochastic Symmetric Programming 42

3.1 The log barrier problem for SSPs 43

3.1.1 Formulation and assumptions 43

3.1.2 Computation of $\nabla_x \tilde{\eta}(\mu, x)$ and $\nabla_{xx}^2 \tilde{\eta}(\mu, x)$ 50
3.2 Self-concordance properties of the log-barrier recourse .......................... 53
  3.2.1 Self-concordance of the recourse function ................................. 53
  3.2.2 Parameters of the self-concordant family ................................. 59
3.3 A class of logarithmic barrier algorithms for solving SSPs ..................... 64
3.4 Complexity analysis ................................................................. 65
  3.4.1 Complexity for short-step algorithm .................................. 66
  3.4.2 Complexity for long-step algorithm .................................. 67

4 A Class of Polynomial Volumetric Barrier Decomposition Algorithms
for Stochastic Symmetric Programming ............................................. 75
  4.1 The volumetric barrier problem for SSPs .................................. 76
    4.1.1 Formulation and assumptions ...................................... 76
    4.1.2 The volumetric barrier problem for SSPs ......................... 77
    4.1.3 Computation of $\nabla_x \eta(\mu, x)$ and $\nabla^2_{xx} \eta(\mu, x)$ ... 80
  4.2 Self-concordance properties of the volumetric barrier recourse .......... 85
    4.2.1 Self-Concordance of $\eta(\mu, \cdot)$ ................................ 86
    4.2.2 Parameters of the self-concordant family ......................... 93
  4.3 A class of volumetric barrier algorithms for solving SSPs .................. 97
  4.4 Complexity analysis ................................................................. 98
    4.4.1 Complexity for short-step algorithm ................................ 100
    4.4.2 Complexity for long-step algorithm ................................ 101

5 Some Applications ........................................................................... 106
  5.1 Two applications of SSOCPs ...................................................... 106
    5.1.1 Stochastic Euclidean facility location problem ..................... 106
    5.1.2 Portfolio optimization with loss risk constraints .................. 112
  5.2 Two applications of SRQCPs ...................................................... 118
    5.2.1 Optimal covering random ellipsoid problem ....................... 118
5.2.2 Structural optimization ........................................ 125

6 Related Open problems: Multi-Order Cone Programming Problems 130

6.1 Multi-order cone programming problems ................................ 131
6.2 Duality ........................................................................ 134
6.3 Multi-order cone programming problems over integers ................. 138
6.4 Multi-order cone programming problems under uncertainty .......... 139
6.5 An application .................................................................. 140
   6.5.1 CERFLPs—An MOCP model ........................................ 142
   6.5.2 DERFLPs—a 0-1MOCP model ...................................... 143
   6.5.3 ERFLPs with integrality constraints—An MIMOCP model .... 145
   6.5.4 Stochastic CERFLPs—An SMOCP model ......................... 146

7 Conclusion ................................................................. 150
## List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CERFLP</td>
<td>continuous Euclidean-rectilinear facility location problem</td>
</tr>
<tr>
<td>CFLP</td>
<td>continuous facility location problem</td>
</tr>
<tr>
<td>DERFLP</td>
<td>discrete Euclidean-rectilinear facility location problem</td>
</tr>
<tr>
<td>DFLP</td>
<td>discrete facility location problem</td>
</tr>
<tr>
<td>DLP</td>
<td>deterministic linear programming</td>
</tr>
<tr>
<td>DRQCP</td>
<td>deterministic rotated quadratic cone programming</td>
</tr>
<tr>
<td>DSDP</td>
<td>deterministic semidefinite programming</td>
</tr>
<tr>
<td>DSOCP</td>
<td>deterministic second-order cone programming</td>
</tr>
<tr>
<td>DSP</td>
<td>deterministic symmetric programming</td>
</tr>
<tr>
<td>EFLP</td>
<td>Euclidean facility location problem</td>
</tr>
<tr>
<td>ERFLP</td>
<td>Euclidean-rectilinear facility location problem</td>
</tr>
<tr>
<td>ESFLP</td>
<td>Euclidean single facility location problem</td>
</tr>
<tr>
<td>FLP</td>
<td>facility location problem</td>
</tr>
<tr>
<td>KKT</td>
<td>Karush-Kuhn-Tucker conditions</td>
</tr>
<tr>
<td>MFLP</td>
<td>multiple facility location problem</td>
</tr>
<tr>
<td>MIMOCP</td>
<td>mixed integer multi-order cone programming</td>
</tr>
<tr>
<td>MOCP</td>
<td>(deterministic) multi-order cone programming</td>
</tr>
<tr>
<td>0 – 1MOCP</td>
<td>0-1 multi-order cone programming</td>
</tr>
<tr>
<td>POCP</td>
<td>$p^{th}$ – order cone programming</td>
</tr>
<tr>
<td>RFLP</td>
<td>rectilinear facility location problem</td>
</tr>
<tr>
<td>SFLP</td>
<td>stochastic facility location problem</td>
</tr>
<tr>
<td>SLP</td>
<td>stochastic linear programming</td>
</tr>
<tr>
<td>SMOCP</td>
<td>stochastic multi-order cone programming</td>
</tr>
<tr>
<td>SRQCP</td>
<td>stochastic rotated quadratic cone programming</td>
</tr>
<tr>
<td>SSDP</td>
<td>stochastic semidefinite programming</td>
</tr>
</tbody>
</table>
SSOCP  stochastic second-order cone programming
SSP  stochastic symmetric programming
Arw(\(x\))  the arrow-shaped matrix associated with the vector \(x\)
Aut(\(\mathcal{K}\))  the automorphism group of a cone \(\mathcal{K}\)
diag(\(\cdot\))  the operator that maps its argument to a block diagonal matrix
\(GL(n, \mathbb{R})\)  the general linear group of degree \(n\) over \(\mathbb{R}\)
int(\(\mathcal{K}\))  the interior of a cone \(\mathcal{K}\)
\(\mathcal{E}^n\)  the \(n\) dimensional real vector space whose elements are indexed from 0
\(\mathcal{E}^n_+\)  the second-order cone of dimension \(n\)
\(\hat{\mathcal{E}}^n_+\)  the rotated quadratic cone of dimension \(n\)
\(\mathcal{H}^n\)  the space of complex Hermitian matrices of order \(n\)
\(\mathcal{H}^n_+\)  the space of complex Hermitian semidefinite matrices of order \(n\)
\(\mathcal{K}_J\)  the cone of squares of a Euclidean Jordan algebra \(J\)
\(Q^n_p\)  the \(p^{th}\)-order cone of dimension \(n\)
\(Q\mathcal{H}^n\)  the space of quaternion Hermitian matrices of order \(n\)
\(Q\mathcal{H}^n_+\)  the space of quaternion Hermitian semidefinite matrices of order \(n\)
\(S^n\)  the space of real symmetric matrices of order \(n\)
\(S^n_+\)  the space of real symmetric positive semidefinite matrices of order \(n\)
\(\mathbb{R}^n\)  the space of real vectors of dimension \(n\)
\(\mathbb{R}^n_+\)  the cone of nonnegative orthants of \(\mathbb{R}^n\)
\(|\|x\||\)  the Frobenius norm of an element \(x\)
\(|\|x\||_2\)  the Euclidean norm of a vector \(x\)
\(|\|x\||_p\)  the \(p\)-norm of a vector \(x\)
\([x]\)  the linear representation of an element \(x\)
\([x]\)  the quadratic representation of an element \(x\)
$X \succeq 0$  the matrix $X$ is positive semidefinite

$X \succ 0$  the matrix $X$ is positive definite

$x \succeq 0$  the vector $x$ lies in a second-order cone of an appropriate dimension

$x \succ 0$  the vector $x$ lies in the interior of a second-order cone of an appropriate dimension

$x \succeq^N 0$  the vector $x$ lies in the Cartesian product of $N$ second-order cones with appropriate dimensions

$x \hat{\succeq} 0$  the vector $x$ lies in a rotated quadratic cone of an appropriate dimension

$x \hat{\succeq}^N 0$  the vector $x$ lies in the Cartesian product of $N$ rotated quadratic cones with appropriate dimensions

$x \succeq_{K_J} 0$  $x$ is an element of a symmetric cone $K_J$

$x \succ_{K_J} 0$  $x$ is an element of the interior of a symmetric cone $K_J$

$x \succeq_{(p)} 0$  the vector $x$ lies in a $p^{th}$-order cone of an appropriate dimension

$x \succeq_{N(p)} 0$  the vector $x$ lies in the Cartesian product of $N$ $p^{th}$-order cones with appropriate dimensions

$x \succeq_{(p_1,p_2,\ldots,p_N)} 0$  the vector $x$ lies in the Cartesian product of $N$ cones of orders $p_1, p_2, \ldots, p_N$ and with appropriate dimensions

$x \circ y$  the Jordan multiplication of elements $x$ and $y$ of a Jordan algebra

$x \bullet y$  the inner product $\text{trace}(x \circ y)$ of elements $x$ and $y$ of a Euclidean Jordan algebra
Chapter 1

Introduction and Background

1.1 Introduction

The purpose of this dissertation is to introduce the two-stage stochastic symmetric programs (SSPs)\(^1\) with recourse and to study this problem in the dual standard form:

\[
\begin{align*}
\text{max} & \quad c^T x + \mathbb{E} \left[ Q(x, \omega) \right] \\
\text{s.t.} & \quad A x + \xi = b \\
& \quad \xi \in K_1,
\end{align*}
\]

(1.1.1)

where \(x\) and \(\xi\) are the first-stage decision variables, and \(Q(x, \omega)\) is the minimum of the problem

\[
\begin{align*}
\text{max} & \quad d(\omega)^T y \\
\text{s.t.} & \quad W(\omega) y + \zeta = h(\omega) - T(\omega) x \\
& \quad \zeta \in K_2,
\end{align*}
\]

(1.1.2)

\(^1\)Based on tradition in optimization literature, we use the term *stochastic symmetric program* to mean the generic form of a problem, and the term *stochastic symmetric programming* to mean the field of activities based on that problem. While both will be denoted by the acronym SSP, the plural of the first usage will be denoted by the acronym SSPs. Acronyms DLP, DRQCP, DSDP, DSOCP, DSP, MIMOCP, MOCP, 0-IMOCP, SLP, SMOCP, SRQCP SSDP, and SSOCP are defined and used accordance with this custom.
where $\mathbf{y}$ and $\mathbf{\zeta}$ are the second-stage variables,  

$$
\mathbb{E}[Q(\mathbf{x}, \omega)] := \int_\Omega Q(\mathbf{x}, \omega) P(d\omega),
$$

the matrix $A$ and the vectors $\mathbf{b}$ and $\mathbf{c}$ are deterministic data, and the matrices $W(\omega)$ and $T(\omega)$ and the vectors $\mathbf{h}(\omega)$ and $\mathbf{d}(\omega)$ are random data whose realizations depend on an underlying outcome $\omega$ in an event space $\Omega$ with a known probability function $P$. The cones $\mathcal{K}_1$ and $\mathcal{K}_2$ are symmetric cones (i.e., closed, convex, pointed, self-dual cones with their automorphism groups acting transitively on their interiors) in $\mathbb{R}^{n_1}$ and $\mathbb{R}^{n_2}$. (Here, $n_1$ and $n_2$ are positive integers.)

The birth of symmetric programming (also known as symmetric cone programming [36]) as a subfield of convex optimization can be dated back to 2003. The main motivation of this generalization was its ability to handle many important applications that cannot be covered by linear programming. In symmetric programs, we minimize a linear function over the intersection of an affine set and a so called symmetric cone. In particular, if the symmetric cone is the nonnegative orthant cone, the result is linear programming, if it is the second-order cone, the result is second-order cone programming, and if it is the semidefinite cone (the cone of all real symmetric positive semidefinite matrices), the result is semidefinite programming. It is seen that symmetric programming is a generalization of linear programming that includes second-order cone programming and semidefinite programming as special cases. We shall refer to such problems as deterministic linear programs (DLPs), deterministic second-order cone programs (DSOCPs), deterministic semidefinite programs (DSDPs) and deterministic symmetric programs (DSPs) because the data defining applications leading to such problems is assumed to be known with certainty. It has been found that DSPs are related to many application areas including finance, geometry, robust linear programming, matrix optimization, norm minimization problems, and relaxations in combinatorial optimization. We refer the reader to the survey papers by Todd [38] and Vandenberghhe and Boyd [41] which discuss, in particular, DSDP and its applications, and the survey papers of Alizadeh and Goldfarb [1], and Lobo, et al. [23] which discuss, in particular, DSOCPs with a number of applications in many areas.
Deterministic optimization problems are formulated to find optimal decisions in problems with certainty in data. In fact, in some applications we cannot specify the model entirely because it depends on information which is not available at the time of formulation but it will be determined at some point in the future. Stochastic programs have been studied since 1950s to handle those problems that involve uncertainty in data. See [17, 32, 33] and references contained therein. In particular, two-stage stochastic linear programs (SLPs) have been established to formulate many applications (see [15] for example) of linear programming with uncertain data. There are efficient algorithms (both interior and noninterior point) for solving SLPs. The class of SSP problems may be viewed as an extension of DSPs (by allowing uncertainty in data) on the one hand, and as an extension of SLPs (where $K_1$ and $K_2$ are both cones of nonnegative orthants) or, more generally, stochastic semidefinite programs (SSDPs) with recourse [9, 25] (where $K_1$ and $K_2$ are both semidefinite cones) on the other hand.

Interior point methods [30] are considered to be one of the most successful classes of algorithms for solving deterministic (linear and nonlinear) convex optimization problems. This provides motivation to investigate whether decomposition-based interior point algorithms can be developed for stochastic programming. Zhao [48] derived a decomposition algorithm for SLPs based on a logarithmic barrier and proved its polynomial complexity. Mehrotra and Özevin [25] have proved important results that extend the work of Zhao [48] to the case of SSDPs including a derivation of a polynomial logarithmic barrier decomposition algorithm for this class of problems that extends Zhao’s algorithm for SLPs. An alternative to the logarithmic barrier is the volumetric barrier of Vaidya [40] (see also [5, 6, 7]). It has been observed [8] that certain cutting plane algorithms [21] for SLPs based on the volumetric barrier perform better in practice than those based on the logarithmic barrier. Recently, Ariyawansa and Zhu [10] have derived a class of decomposition algorithms for SSDPs based on a volumetric barrier analogous to work of Mehrotra and

Concerning algorithms for DSPs, Schmieta and Alizadeh [35, 36] and Rangarajan [34] have extended interior point algorithms for DSDP to DSP. Concerning algorithms for SSPs, we know of no interior point algorithms for solving them that exploits the special structure of the symmetric cone as it is done in [35, 36, 34] for DSP. The question that naturally arises now is whether interior point methods could be derived for solving SSPs, and if the answer is affirmative, it is important to ask whether or not we can prove the polynomial complexity of the resulting algorithms. Our particular concern in this dissertation is to extend decomposition-based interior point methods for SSDPs to stochastic optimization problems over all symmetric cones based on both logarithmic and volumetric barriers and also to prove the polynomial complexity of the resulting algorithms. In fact, there is a unifying theory [18] based on Euclidean Jordan algebras that connects all symmetric cones. This theory is very important for a sound understanding of the Jordan algebraic characterization of symmetric cones and, hence, a correct understanding of the very close equivalence between Problem (1.1.1, 1.1.2) and our constructive definition of an SSP which will be presented in Chapter 2.

Now let us indicate briefly how to solve Problem (1.1.1, 1.1.2). We assume that the event space \( \Omega \) is discrete and finite. In practice, the case of interest is when the random data would have a finite number of realizations, because the inputs of the modeling process require that SSPs be solved for a finite number of scenarios. The general scheme of the decomposition-based interior point algorithms is as follows. As we have an SSP with a finite scenarios, we can explicitly formulate the problem as a large scale DSP, we then use primal-dual interior point methods to solve the resulting large scale formulation directly, which can be successfully accomplished by utilizing the special structure of the underlying symmetric cones.

This dissertation is organized as follows. In the remaining part of this chapter, we outline a minimal foundation of the theory of Euclidean Jordan algebras. Based on Eu-
clidean Jordan algebras. In Chapter 2 we explicitly introduce the constructive definition of SSP problem in both the primal and dual standard forms and then introduce six general classes of optimization problems that can be cast as SSPs. The focus of Chapter 3 is on extending the work of Mehrotra and Özevin [25] to the case of SSPs by deriving a class of logarithmic barrier decomposition algorithms for SSPs and establishing the polynomial complexity of the resulting algorithm.

In Chapter 4, we extend the work of Ariyawansa and Zhu [10] to the case of SSPs by deriving a class of volumetric barrier decomposition algorithms for SSPs and proving polynomial complexity of certain members of the class of algorithms.

Chapter 5 is devoted to describe four applications leading to, indeed, two important special cases of SSPs. More specifically, we describe the stochastic Euclidean facility location problem and the portfolio optimization problem with loss risk constraints as two applications of SSPs when the symmetric cones $K_1$ and $K_2$ are both second-order cones, and then describe the optimal covering random ellipsoid problem and an application in structural optimization as two applications of SSPs when the symmetric cones $K_1$ and $K_2$ are both rotated quadratic cones (see §2.2 for definitions).

The material in Chapters 3-5 are independent, but they essentially depend on the material of Chapters 1 and 2. So, after reading Chapter 2, one can proceed directly to Chapter 3 and/or Chapter 4 concerning theoretical results (algorithms and complexity analysis), and/or to Chapter 5 concerning applicational models (see Figure 1.1).

In Chapter 6, we propose the so called multi-order cone programming problem as a new conic optimization problem. This problem is beyond the scope of this dissertation because it is over non-symmetric cones, so we will leave it as an unsolved open problem. We indicate weak and strong duality relations of this optimization problem and describe an application of it.

We conclude this dissertation in Chapter 7 by summarizing its contributions and indicating some possible future research directions.
We gave a concise definition for the symmetric cones immediately after Problem (1.1.1, 1.1.2). In §1.2, we write down this definition more explicitly, and in §1.3 we review the theory of Euclidean Jordan algebras that connects all symmetric cones. The text of Faraut and Korányi [18] covers the foundations of this theory.

In Chapter 2, the reader will see how general and abstract the problem is. In order to make our presentation concrete, we will describe two interesting examples of symmetric cones with some details throughout this chapter: the second-order cone and the cone of real symmetric positive semidefinite matrices. Our presentation in §1.2 and §1.3 mostly follows that of [18] and [36], and, in particular, most examples in §1.3 are taken from [36].

We now introduce some notations that will be used in the sequel. We use $\mathbb{R}$ to denote the field of real numbers. If $A \subseteq \mathbb{R}^k$ and $B \subseteq \mathbb{R}^l$, then the Cartesian product of $A \times B := \{(x; y) : x \in A \text{ and } y \in B\}$. We use $\mathbb{R}^{m \times n}$ to denote the vector spaces of real $m \times n$ matrices. The matrices $0_n, I_n \in \mathbb{R}^{n \times n}$ denote, respectively, the zero and the identity matrices of order $n$ (we will write $0_n$ and $I_n$ as 0 and I when $n$ is known from the context). All vectors we use are column vectors with superscript “T” indicating transposition. We use “,” for adjoining vectors and matrices in a row, and use “;” for adjoining them in a column. So, for example, if $x, y,$ and $z$ are vectors, we have:
\[
\begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix}
= (x^T, y^T, z^T)^T = (x; y; z).
\]

For each vector \( \mathbf{x} \in \mathbb{R}^n \) whose first entry is indexed with 0, we write \( \bar{\mathbf{x}} \) for the subvector consisting of entries 1 through \( n - 1 \); therefore \( \mathbf{x} = (x_0; \bar{x}) \in \mathbb{R} \times \mathbb{R}^{n-1} \). We let \( \mathcal{E}^n \) denote the \( n \) dimensional real vector space \( \mathbb{R} \times \mathbb{R}^{n-1} \) whose elements \( \mathbf{x} \) are indexed with 0, and denote the space of real symmetric matrices of order \( n \) by \( \mathcal{S}^n \).

### 1.2 What is a symmetric cone?

This section and the next section are elementary. We start with some basic definitions finally leading us to the definition of the symmetric cone.

Let \( V \) be a finite-dimensional Euclidean vector space over \( \mathbb{R} \) with inner product \( \langle \cdot, \cdot \rangle \). A subset \( S \) of \( V \) is said to be **convex** if it is closed with respect to convex combinations of finite subsets of \( S \), i.e., for any \( \lambda \in (0, 1), \quad \mathbf{x}, \mathbf{y} \in S \) implies that \( \lambda \mathbf{x} + (1 - \lambda) \mathbf{y} \in S \). A subset \( K \subset V \) is said to be a **cone** if it is closed under scalar multiplication by positive real numbers, i.e., if for any \( \lambda > 0, \quad \mathbf{x} \in K \) implies that \( \lambda \mathbf{x} \in K \). A **convex cone** is a cone that is also a convex set. A cone is said to be **closed** if it is closed with respect to taking limits, **solid** if it has a nonempty interior, **pointed** if it contains no lines or, alternatively, it does not contain two opposite nonzero vectors (so the origin is an extreme point), and **regular** if it is a closed, convex, pointed, solid cone.

By \( GL(n, \mathbb{R}) \) the general linear group of degree \( n \) over \( \mathbb{R} \) (i.e., the set of \( n \times n \) invertible matrices with entries from \( \mathbb{R} \), together with the operation of ordinary matrix multiplication). For a regular cone \( K \subset V \), we denote by \( \text{int}(K) \) the interior of \( K \), and by \( \text{Aut}(K) \) the automorphism group of \( K \), i.e., \( \text{Aut}(K) := \{ \varphi \in GL(n, \mathbb{R}) : \varphi(K) = K \} \).

**Definition 1.2.1.** Let \( V \) be a finite-dimensional real Euclidean space. A regular \( K \subset V \)
is said to be *homogeneous* if for each \( u, v \in \text{int}(\mathcal{K}) \), there exists an invertible linear map \( \varphi : \mathcal{V} \rightarrow \mathcal{V} \) such that

1. \( \varphi(\mathcal{K}) = \mathcal{K} \), i.e., \( \varphi \) is an automorphism of \( \mathcal{K} \), and

2. \( \varphi(u) = v \).

In other words, \( \text{Aut}(\mathcal{K}) \) acts transitively on the interior of \( \mathcal{K} \).

A regular \( \mathcal{K} \subset \mathcal{V} \) said to be *self-dual* if it coincides with its dual cone \( \mathcal{K}^* \), i.e.,

\[
\mathcal{K} = \mathcal{K}^* := \{ s \in \mathcal{V} : \langle x, s \rangle \geq 0, \forall x \in \mathcal{K} \},
\]

and *symmetric* if it is both homogeneous and self-dual.

Almost all conic optimization problems in real world applications are associated with symmetric cones such as the nonnegative orthant cone, the second-order cone (see below for definitions), and the cone of positive semi-definite matrices over the real or complex numbers. Except for Chapter 7 which proposes an unsolved open optimization problem over a non-symmetric cone, our focus in this dissertation is on those cones that are symmetric.

The material in the rest of this section is *not* needed later because the theory of Jordan algebras will lead us independently to the same results. However, we include this material for the sake of clarity of the examples and completeness.

**Example 1. The second-order cone \( \mathcal{E}^n_+ \).**

We show that the second-order cone (also known as the quadratic, Lorentz, or the ice cream cone) of dimension \( n \)

\[
\mathcal{E}^n_+ := \{ \xi \in \mathcal{E}^n : \xi_0 \geq \|\xi\|_2 \},
\]

with the standard inner product, \( \langle \xi, \zeta \rangle := \xi^T \zeta \), is a symmetric cone.
In Lemma 6.2.1, we prove that the dual cone of the $p^{th}$-order cone is the $q^{th}$-order cone, i.e.,
\[
\{ \xi \in \mathcal{E}^n : \xi_0 \geq \|\bar{\xi}\|_p \}^* = \{ \xi \in \mathcal{E}^n : \xi_0 \geq \|\bar{\xi}\|_q \},
\]
where $1 \leq p \leq \infty$ and $q$ is conjugate to $p$ in the sense that $1/p + 1/q = 1$. Picking $p = 2$ (hence $q = 2$) implies that $(\mathcal{E}_+^n)^* = \mathcal{E}_+^n$. This demonstrates the self-duality of $\mathcal{E}_+^n$. So, to prove that the cone $\mathcal{E}_+^n$ is symmetric, we only need to show that it is homogeneous. The proof follows that of Example 1 in §2 of [18, Chapter I].

First, notice that $\mathcal{E}_+^n$ can be redefined as
\[
\mathcal{E}_+^n := \{ \xi \in \mathcal{E}^n : \xi^T J \xi \geq 0, \xi_0 \geq 0 \},
\]
where
\[
J := \begin{bmatrix} 1 & 0^T \\ 0 & -I_{n-1} \end{bmatrix}.
\]
Notice also that each element of the group $\mathcal{G} := \{ A \in \mathbb{R}^{n \times n} : A^T J A = J \}$ maps $\mathcal{E}_+^n$ onto itself (because, for every $A \in \mathcal{G}$, we have that $(A \xi)^T J (A \xi) = \xi^T J \xi$), and so does the direct product $\mathcal{H} := \mathbb{R}_+ \times \mathcal{G}$. It now remains to show that the group $\mathcal{H}$ acts transitively on the interior of $\mathcal{E}_+^n$. To do so, it is enough show that, for any $x \in \text{int}(\mathcal{E}_+^n)$, there exists an element in $\mathcal{H}$ that maps $e$ to $x$, where $e := (1; 0) \in \mathcal{E}^n$.

Note that we may write $x$ as $x = \lambda y$ with $\lambda = \sqrt{x^T J x}$ and $y \in \mathcal{E}^n$. Moreover, there exists a reflector matrix $Q$ such that $\bar{y} = Q (0; \ldots; 0; r)$, with $r = \|\bar{y}\|_2$. We then have
\[
y_0^2 - r^2 = y_0^2 - \|\bar{y}\|_2^2 = y^T J y = \frac{1}{\lambda^2} x^T J x = 1.
\]
Therefore, there exists $t \geq 0$ such that $y_0 = \cosh t$ and $r = \sinh t$.  

We define
\[ \hat{Q} := \begin{bmatrix} 1 & 0^T \\ 0 & Q \end{bmatrix} \quad \text{and} \quad H_t := \begin{bmatrix} \cosh t & 0^T & \sinh t \\ 0 & I_{n-2} & 0 \\ \sinh t & 0^T & \cosh t \end{bmatrix}. \]
Observing that \( \hat{Q}, H_t \in \mathcal{G} \) yields that \( \hat{Q}H_t \in \mathcal{G} \), and therefore \( \lambda \hat{Q}H_t \in \mathcal{H} \). The result follows by noting that \( x = \lambda \hat{Q}H_t \mathbf{e} \).

**Example 2. The cone of real symmetric positive semidefinite matrices \( S^+_n \).**
The residents of the interior of the cone of real symmetric positive semidefinite matrices of order \( n \), \( S^+_n \), are the positive definite matrices, the residents of the boundary of this cone are the singular positive semidefinite matrices (having at least one 0 eigenvalue), and the only matrix that resides at the origin is the positive semidefinite matrix having all 0 eigenvalues. We now show that \( S^+_n \), with the Frobenius inner product \( \langle U, V \rangle := \text{trace}(UV) \), is a symmetric cone.

We first verify that \( S^+_n \) is self-dual, i.e.,
\[ S^+_n = (S^+_n)^* = \{ U \in S^n : \text{trace}(UV) \geq 0 \text{ for all } V \in S^+_n \}. \]
We can easily prove that \( S^+_n \subseteq (S^+_n)^* \) by assuming that \( X \in S^+_n \) and observing that, for any \( Y \in S^+_n \), we have
\[ \text{trace}(XY) = \text{trace}(X^{\frac{1}{2}}YYX^{\frac{1}{2}}) \geq 0. \]
Here we used the positive definiteness of \( X^{\frac{1}{2}}YYX^{\frac{1}{2}} \) to obtain the inequality. In fact, any \( U \in S^n \) can be written as \( U = Q\Lambda Q^T \) where \( Q \in \mathbb{R}^{n\times n} \) is an orthogonal matrix and \( \Lambda \in \mathbb{R}^{n\times n} \) is a diagonal matrix whose diagonal entries are the eigenvalues of \( U \) (see for example Watkins [43, Theorem 5.4.20]). Therefore, if \( U \in S^+_n \) we have that
\[ \text{trace}(U) = \text{trace}(Q\Lambda Q^T) = \text{trace}(\Lambda QQ^T) = \text{trace}(\Lambda) = \sum_{i=1}^{n} \lambda_i \geq 0. \]
To prove that $(S^n_+)^* \subseteq S^n_+$, assume that $X \notin S^n_+$, then there exists a nonzero vector $y \in \mathbb{R}^n$ such that $\text{trace}(Xyy^T) = y^TXy < 0$, which shows that $X \notin (S^n_+)^*$.

So, to prove that the cone $S^n_+$ is symmetric, it remains to show that it is homogeneous. The proof follows exactly that of Example 2 in §2 of [18, Chapter I].

For $P \in \text{GL}(n, \mathbb{R})$, we define a linear map $\varphi_P : S^n \rightarrow S^n$ by

$$\varphi_P(X) = PXP^T.$$  

Note that $\varphi_P$ maps $S^n_+$ into itself. By the Cholesky decomposition theorem, if $X \in \text{int}(S^n_+)$ then $X$ can be decomposed into a product $X = LL^T = \varphi_L(I_n)$, where $L \in \text{GL}(n, \mathbb{R})$, which establishes the result.

### 1.3 Symmetric cones and Euclidean Jordan algebras

In this section we give a review some of the basic definitions and theory of Euclidean Jordan algebras that are necessary for our subsequent development. We will also see how we can use Euclidean Jordan algebras to obtain symmetric cones.

Let $\mathcal{J}$ be a finite-dimensional vector space over $\mathbb{R}$. A map $\circ : \mathcal{J} \times \mathcal{J} \rightarrow \mathcal{J}$ is called **bilinear** if for all $x, y \in \mathcal{J}$ and for all $\alpha, \beta \in \mathbb{R}$, we have that $(\alpha x + \beta y) \circ z = \alpha(x \circ z) + \beta(y \circ z)$ and $x \circ (\alpha y + \beta z) = \alpha(x \circ y) + \beta(x \circ z)$.

**Definition 1.3.1.** A finite-dimensional vector space $\mathcal{J}$ over $\mathbb{R}$ is called an **algebra** over $\mathbb{R}$ if a bilinear map $\circ : \mathcal{J} \times \mathcal{J} \rightarrow \mathcal{J}$ exists.

Let $x$ be an element in an algebra $\mathcal{J}$, then for $n \geq 2$ we define $x^n$ recursively by $x^n := x \circ x^{n-1}$.

**Definition 1.3.2.** Let $\mathcal{J}$ be a finite-dimensional $\mathbb{R}$ algebra with a bilinear product $\circ : \mathcal{J} \times \mathcal{J} \rightarrow \mathcal{J}$. Then $(\mathcal{J}, \circ)$ is called a **Jordan algebra** if for all $x, y \in \mathcal{J}$...
1. \( x \circ y = y \circ x \) (commutativity);

2. \( x \circ (x^2 \circ y) = x^2 \circ (x \circ y) \) (Jordan’s axiom).

The product \( x \circ y \) between two elements \( x \) and \( y \) of a Jordan algebra \((J, \circ)\) is called the Jordan multiplication between \( x \) and \( y \). A Jordan algebra \((J, \circ)\) has an identity element if there exists a (necessarily unique) element \( e \in J \) such that \( x \circ e = e \circ x = x \) for all \( x \in J \). A Jordan algebra \((J, \circ)\) is not necessarily associative, that is, \( x \circ (y \circ z) = (x \circ y) \circ z \) may not hold in general. However, it is power associative, i.e., \( x^p \circ x^q = x^{p+q} \) for all integers \( p, q \geq 1 \).

Example 3. It is easy to see that the space \( \mathbb{R}^{n \times n} \) of \( n \times n \) real matrices with the Jordan multiplication \( X \circ Y := (XY + YX)/2 \) forms a Jordan algebra with identity \( I_n \).

Example 4. It can be verified that the space \( \mathcal{E}^n \) with the Jordan multiplication \( x \circ y := (x^T y; x_0 y + y_0 x) \) forms a Jordan algebra with the identity \((1; 0) \in \mathcal{E}^n \).

Definition 1.3.3. A Jordan algebra \( J \) is called Euclidean if there exists an inner product \( \langle \cdot, \cdot \rangle \) on \((J, \circ)\) such that for all \( x, y, z \in J \)

1. \( \langle x, x \rangle > 0 \ \forall \ x \neq 0 \) (positive definiteness);

2. \( \langle x, y \rangle = \langle y, x \rangle \) (symmetry);

3. \( \langle x, y \circ z \rangle = \langle x \circ y, z \rangle \) (associativity).

That is, \( J \) admits a positive definite, symmetric, quadratic form which is also associative.

In the sequel, we consider only Euclidean Jordan algebras with identity. We simply denote the Euclidean Jordan algebra \((J, \circ)\) by \( J \).

Example 5. The space \( \mathbb{R}^{n \times n} \) is not a Euclidean Jordan algebra. However, under the operation “\( \circ \)” defined in Example 3, the subspace \( \mathcal{S}^n \) is a Jordan subalgebra of \( \mathbb{R}^{n \times n} \), and,
indeed, is a Euclidean Jordan algebra with the inner product $\langle X, Y \rangle = \text{trace}(X \circ Y) = \text{trace}(XY)$. While both the symmetry and the associativity can be easily proved by using the fact that $\text{trace}(XY) = \text{trace}(YX)$, the positive definiteness can be immediately obtained by observing that $\text{trace}(X^2) > 0$ for $X \neq 0$.

**Example 6.** It is easy to verify that the space $\mathcal{E}^n$ (with “$\circ$” defined as in Example (4)) is a Euclidean Jordan algebra with the inner product $\langle x, y \rangle = x^\top y$.

Many of the results below also hold for the general setting of Jordan algebras, but here we focus entirely on Euclidean Jordan algebras with identity as that generality is not needed for our subsequent development.

Since a Euclidean Jordan algebra $\mathcal{J}$ is power associative, we can define the concepts of rank, minimal and characteristic polynomials, eigenvalues, trace, and determinant for it.

**Definition 1.3.4.** Let $\mathcal{J}$ be a Euclidean Jordan algebra. Then

1. for $x \in \mathcal{J}$, $\deg(x) := \min \{ r > 0 : \{ e, x, x^2, \ldots, x^r \} \text{ is linearly dependent} \}$ is called the *degree* of $x$;

2. $\text{rank}(\mathcal{J}) := \max_{x \in \mathcal{J}} \deg(x)$ is called the *rank* of $\mathcal{J}$.

Let $x$ be an element of degree $d$ in a Euclidean Jordan algebra $\mathcal{J}$. We define $\mathbb{R}[x]$ as the set of all polynomials in $x$ over $\mathbb{R}$:

$$\mathbb{R}[x] := \left\{ \sum_{i=0}^{\infty} a_i x^i : a_i \in \mathbb{R}, a_i = 0 \text{ for all but a finite number of } i \right\}.$$ 

Since $\{ e, x, x^2, \ldots, x^d \}$ is linearly dependent, there exists a nonzero monic polynomial $q \in \mathbb{R}[x]$ of degree $d$, such that $q(x) = 0$. In other words, there are real numbers $a_1(x), a_2(x), \ldots, a_d(x)$, not all zero, such that

$$q(x) := x^d - a_1(x)x^{d-1} + a_2(x)x^{d-2} + \cdots + (-1)^da_d(x)e = 0.$$
Clearly $q$ is of minimal degree of those polynomials in $\mathbb{R}[x]$ which have the above properties, so we call it (or, alternatively, the polynomial $p(\lambda) := \lambda^d - a_1(x)\lambda^{d-1} + a_2(x)\lambda^{d-2} + \cdots + (-1)^d a_d(x)$) the minimal polynomial of $x$. Note that the minimal polynomial of an element $x \in J$ is unique, because otherwise, as its monic, we will have a contradiction to the minimality of its degree.

An element $x \in J$ is called regular if $\deg(x) = \text{rank}(J)$. If $x$ is a regular element of a Euclidean Jordan algebra, then we define its characteristic polynomial to be equal to its minimal polynomial. We have the following proposition.

**Proposition 1.3.1** ([18, Proposition II.2.1]). Let $J$ be an algebra with rank $r$. The set of regular elements is open and dense in $J$. There exist polynomials $a_1, a_2, \ldots, a_r$ such that the minimal polynomial of every regular element $x$ is given by $p(\lambda) = \lambda^r - a_1(x)\lambda^{r-1} + a_2(x)\lambda^{r-2} + \cdots + (-1)^r a_r(x)$. The polynomials $a_1, a_2, \ldots, a_r$ are unique and $a_i$ is homogeneous with degree $i$.

The polynomial $p(\lambda)$ is called the characteristic polynomial of the regular element $x$. Since the set of regular elements are dense in $J$, by continuity we can extend characteristic polynomials to all $x$ in $J$. So, the minimal polynomial coincides with the characteristic polynomial for regular elements and divides the characteristic polynomial of non-regular elements.

**Definition 1.3.5.** Let $x$ be an element in a rank-$r$ algebra $J$, then its eigenvalues are the roots $\lambda_1, \lambda_2, \ldots, \lambda_r$ of its characteristic polynomial $p(\lambda) = \lambda^r - a_1(x)\lambda^{r-1} + a_2(x)\lambda^{r-2} + \cdots + (-1)^r a_r(x)$.

Whereas the minimal polynomial has only simple roots, it is possible, in the case of non-regular elements, that the characteristic polynomial have multiple roots. Indeed, the characteristic and minimal polynomials have the same set of roots, except for their multiplicities. In fact, we can define the degree of $x$ to be the number of distinct eigenvalues of $x$.  

14
Definition 1.3.6. Let $x$ be an element in a rank-$r$ algebra $J$, and $\lambda_1, \lambda_2, \ldots, \lambda_r$ be the roots of its characteristic polynomial $p(\lambda) = \lambda^r - a_1(x)\lambda^{r-1} + a_2(x)\lambda^{r-2} + \cdots + (-1)^ra_r(x)$. Then

1. trace$(x) := \lambda_1 + \lambda_2 + \cdots + \lambda_r = a_1(x)$ is the trace of $x$ in $J$;

2. det$(x) := \lambda_1\lambda_2\cdots\lambda_r = a_r(x)$ is the determinant of $x$ in $J$.

Example 7. All these concepts (characteristic polynomials, eigenvalues, trace, determinant, etc.) are the corresponding concepts used in $S^n$. Observe that rank$(S^n) = n$ because the deg$(X)$ is the number of distinct eigenvalues of $X$ which is, indeed, at most $n$.

Example 8. We can easily prove the following quadratic identity for $x \in \mathcal{E}^n$:

$$x^2 - 2x_0x + (x_0^2 - \|\bar{x}\|_2^2)e = 0.$$ 

Thus, we can define the polynomial $p(\lambda) := \lambda^2 - 2x_0\lambda + (x_0^2 - \|\bar{x}\|_2^2)$ as the characteristic polynomial of $x$ in $\mathcal{E}^n$ and its two roots, $\lambda_{1,2} = x_0 \pm \|\bar{x}\|_2$, are the eigenvalues of $x$. We also have that trace$(x) = 2x_0$ and det$(x) = x_0^2 - \|\bar{x}\|_2^2$. Observe also that $\lambda_1 = \lambda_2$ if and only if $\bar{x} = 0$, and therefore $x$ is a multiple of the identity. Thus, every $x \in \mathcal{E}^n - \{\alpha e : \alpha \in \mathbb{R}\}$ has degree 2. This implies that rank$(\mathcal{E}^n) = 2$, which is, unlike the rank of $S^n$, independent of the dimension of the underlying vector space.

For an element $x \in J$, let $[x] : J \rightarrow J$ be the linear map defined by $[x]y := x \circ y$, for all $y \in J$. Note that $[x]e = x$ and $[x]x = x^2$. Note also that the operators $[x]$ and $[x^2]$ commute, because, by Jordan’s Axiom, $[x][x^2]y = [x^2][x]y$.

Example 9. An equivalent way of dealing with symmetric matrices is dealing with vectors obtained from the vectorization of symmetric matrices. The operator vec : $S^n \rightarrow \mathbb{R}^{n^2}$ creates a column vector from a matrix $X$ by stacking its column vectors below one another.
Note that
\[
\text{vec}(X \circ Y) = \text{vec} \left( \frac{XY + YX}{2} \right) = \frac{1}{2} \left( \text{vec}(XYI) + \text{vec}(IYX) \right) = \frac{1}{2} (I \otimes X + X \otimes I) \text{vec}(Y),
\]
where we used the fact that \(\text{vec}(ABC) = (C^T \otimes A)\text{vec}(B)\) to obtain the last equality (here, the operator \(\otimes : \mathbb{R}^{m \times n} \times \mathbb{R}^{k \times l} \rightarrow \mathbb{R}^{mk \times nl}\) is the Kronecker product which maps the pair of matrices \((A, B)\) into the matrix \(A \otimes B\) whose \((i, j)\) block is \(a_{ij}B\) for \(i = 1, 2, \ldots, m\) and \(j = 1, 2, \ldots, n\)). This gives the explicit formula of the \(\lfloor \cdot \rfloor\) operator for \(S^n\).

**Example 10.** The explicit formula of the \(\lfloor \cdot \rfloor\) operator for \(E^n\) can be immediately given by considering the following multiplication:
\[
x \circ y = \begin{bmatrix} x^T y \\ x_0 \bar{y} + y_0 \bar{x} \end{bmatrix} = \begin{bmatrix} x_0 & \bar{x}^T \\ \bar{x} & x_0 I \end{bmatrix} y.
\]

Here \(A_{rw}(x) \in S^n\) is the arrow-shaped matrix associated with the vector \(x \in E^n\).

For \(x, y \in J\), let \([x, y] : J \times J \rightarrow J\) be the quadratic operator defined by
\[
[x, y] := \lfloor x \rfloor \lfloor y \rfloor + \lfloor y \rfloor \lfloor x \rfloor - [x \circ y].
\]

Therefore, in addition to \([x]\), we can define another linear map \([x] : J \rightarrow J\) associated with \(x\) that is called the quadratic representation and defined by
\[
[x] := 2\lfloor x \rfloor^2 - \lfloor x^2 \rfloor = [x, x].
\]
Example 11. Continuing Example 9 we have

\[
\begin{align*}
[X, Z]\text{vec}(Y) &= [X][Z]\text{vec}(Y) + [Z][X]\text{vec}(Y) - [X \circ Z]\text{vec}(Y) \\
&= \text{vec}(X \circ (Z \circ Y)) + \text{vec}(Z \circ (X \circ Y)) - \text{vec}((X \circ Z) \circ Y) \\
&= \text{vec} \left( \frac{XZY + XYZ + ZYX + YZX}{4} \right) + \text{vec} \left( \frac{ZXY + YZX + XYZ + YXZ}{4} \right) \\
&- \text{vec} \left( \frac{XZY + ZXY + YZX + YXZ}{4} \right) \\
&= \frac{1}{2}(\text{vec}(XYZ) + \text{vec}(ZYX)) \\
&= \frac{1}{2}(X \otimes Z + Z \otimes X)\text{vec}(Y).
\end{align*}
\]

This shows that \([X, Z] := \frac{1}{2}(X \otimes Z + Z \otimes X)\), and, in particular, \([X] := X \otimes X\).

Example 12. Continuing Example 10, we can easily verify that

\[
[x] := 2Arw^2(x) - Arw(x^2) = \begin{bmatrix} ||x||_2^2 & 2x_0\bar{x}^T \\ 2x_0\bar{x} & \det(x)I + 2\bar{x}\bar{x}^T \end{bmatrix}.
\]

Notice that \([e] = [\bar{e}] = I\), \(\text{trace}(e) = r\), \(\det(e) = 1\) (since all the eigenvalues of \(e\) are equal to one). Notice also that the linear operator \([x]\) is symmetric with respect to \((\cdot, \cdot)\), because, by the associativity of the inner product \((\cdot, \cdot)\), we have that \(\langle[x]y, z \rangle = \langle x \circ y, z \rangle = \langle x, y \circ z \rangle = \langle y, [x]z \rangle\). This implies that \([x]\) is also symmetric with respect to \((\cdot, \cdot)\). It is also easy to see that \([x, y + z] = [x, y] + [x, z]\), and consequently \([x, y + z] = [x, y] + [x, z]\).

As the operator \([X]\) in Example 11 plays an important role in the development of the interior point methods for DSDP (SSDP) and the operator in Example 12 plays an important role in the development of the interior point methods for DSOC (SSOCP), we would expect that the operator \([x]\) will play a similar role in the development of the interior point methods for DSP (SSP).

A spectral decomposition is a decomposition of \(x\) into idempotents together with the eigenvalues. Recall that two elements \(c_1, c_2 \in \mathcal{J}\) are said to be orthogonal if \(c_1 \circ c_2 = 0\).
A set of elements of $\mathcal{J}$ is orthogonal if all its elements are mutually orthogonal to each other. An element $c \in \mathcal{J}$ is said to be an idempotent if $c^2 = c$. An idempotent is primitive if it is non-zero and cannot be written as a sum of two (necessarily orthogonal) non-zero idempotents.

**Definition 1.3.7.** Let $\mathcal{J}$ be a Euclidean Jordan algebra. Then a subset $\{c_1, c_2, \ldots, c_r\}$ of $\mathcal{J}$ is called:

1. a complete system of orthogonal idempotents if it is an orthogonal set of idempotents where $c_1 + c_2 + \cdots + c_r = e$;

2. a Jordan frame if it is a complete system of orthogonal primitive idempotents.

**Example 13.** Let $\{q_1, q_2, \ldots, q_n\}$ be an orthonormal subset (all its vectors are mutually orthogonal and all of unit norm (length)) of $\mathbb{R}^n$. Then the set $\{q_1 q_1^T, q_2 q_2^T, \ldots, q_n q_n^T\}$ is a Jordan frame in $\mathcal{S}^n$. In fact, by the orthonormality, we have that

\[
(q_i q_i^T)(q_j q_j^T) = \begin{cases} 
0_n, & \text{if } i \neq j \\
q_i q_i^T, & \text{if } i = j.
\end{cases}
\]

and that $\sum_{i=1}^n q_i q_i^T = I_n$.

**Example 14.** Let $x$ be a vector in $\mathcal{E}^n$. It is easy to see that the set $\left\{\frac{1}{2} \left(1; \frac{x}{\|x\|_2}\right), \frac{1}{2} \left(1; -\frac{x}{\|x\|_2}\right)\right\}$ is a Jordan frame in $\mathcal{E}^n$.

**Theorem 1.3.1** (Spectral decomposition (I), [18]). Let $\mathcal{J}$ be a Euclidean Jordan algebra with rank $r$. Then for $x \in \mathcal{J}$ there exist real numbers $\lambda_1, \lambda_2, \ldots, \lambda_r$ and a Jordan frame $c_1, c_2, \ldots, c_r$ such that $x = \lambda_1 c_1 + \lambda_2 c_2 + \cdots + \lambda_r c_r$, and $\lambda_1, \lambda_2, \ldots, \lambda_r$ are the eigenvalues of $x$.

It is immediately seen that the eigenvalues of elements of Euclidean Jordan algebras are always real, which is not the case for non-Euclidean Jordan algebras.
Example 15. It is known that for any $X \in S^n$ there exists an orthogonal matrix $Q \in \mathbb{R}^{n \times n}$ and a diagonal matrix $\Lambda \in \mathbb{R}^{n \times n}$ such that $X = Q \Lambda Q^T$. In fact, $\lambda_1, \lambda_2, \ldots, \lambda_n$; the diagonal entries of $\Lambda$, and $q_1, q_2, \ldots, q_n$; the columns of $Q$, can be used to rewrite $X$ equivalently as

$$X = \lambda_1 q_1 q_1^T + \lambda_2 q_2 q_2^T + \cdots + \lambda_n q_n q_n^T.$$ 

which, in view of Example 13, gives the spectral decomposition (I) of $X$ in $S^n$.

Example 16. Using Example 14, the spectral decomposition (I) of $x$ in $E^n$ can be obtained by considering the following identity:

$$x = \left( x_0 + \| \bar{x} \|_2 \right) \frac{1}{2} \begin{pmatrix} 1 & \bar{x} \end{pmatrix} e_1 + \left( x_0 - \| \bar{x} \|_2 \right) \frac{1}{2} \begin{pmatrix} 1 & - \bar{x} \end{pmatrix} e_2.$$ 

Theorem 1.3.2 (Spectral decomposition (II), [18]). Let $J$ be a Euclidean Jordan algebra. Then for $x \in J$ there exist unique real numbers $\lambda_1, \lambda_2, \ldots, \lambda_k$, all distinct, and a unique complete system of orthogonal idempotents $c_1, c_2, \ldots, c_k$ such that $x = \lambda_1 c_1 + \lambda_2 c_2 + \cdots + \lambda_k c_k$.

Continuing Examples 15 and 16, we have the following two examples [36].

Example 17. To write the spectral decomposition (II) of $X$ in $S^n$, let $\lambda_1 > \lambda_2 > \cdots > \lambda_k$ be distinct eigenvalues of $X$ such that, for each $i = 1, 2, \ldots, k$, the eigenvalue $\lambda_i$ has multiplicity $m_i$ and orthogonal eigenvectors $q_{i1}, q_{i2}, \ldots, q_{im_i}$. Then $X$ can be written as

$$X = \lambda_1 \sum_{j=1}^{m_1} q_{1j} q_{1j}^T + \lambda_2 \sum_{j=1}^{m_2} q_{2j} q_{2j}^T + \cdots + \lambda_k \sum_{j=1}^{m_k} q_{kj} q_{kj}^T,$$

where the set $\{C_1, C_2, \ldots, C_k\}$ is an orthogonal system of idempotents and $\sum_{i=1}^k C_i = I_n$. Notice that for each eigenvalue $\lambda_i$, the matrix $C_i$ is uniquely written, even though the
corresponding eigenvectors $q_{i1}, q_{i2}, \ldots, q_{im}$ may not be unique.

**Example 18.** By Example 16, the eigenvalues of $x \in E^n$ are $\lambda_{1,2} = x_0 \pm \|x\|_2$, and therefore, as mentioned earlier, only multiples of identity have multiples eigenvalues. In fact, if $x = \alpha e$, for some $\alpha \in \mathbb{R}$, then its spectral decomposition (II) is simply $\alpha e$ (here $\{e\}$ is the singleton system of orthonormal idempotents).

**Definition 1.3.8.** Let $J$ be a Euclidean Jordan algebra. We say that $x \in J$ is **positive semidefinite** (**positive definite**) if all its eigenvalues are nonnegative (**positive**).

**Proposition 1.3.2** ([18, Proposition III.2.2]). If the elements $x$ and $y$ are positive definite, then the element $|x|y$ is so.

**Definition 1.3.9.** We say that two elements $x$ and $y$ of a Euclidean Jordan algebra are **simultaneously decomposed** if there is a Jordan frame $\{c_1, c_2, \ldots, c_r\}$ such that $x = \sum_{i=1}^r \lambda_i c_i$ and $y = \sum_{i=1}^r \mu_i c_i$.

For $x \in J$, it is now possible to rewrite the definition of $x^2$ as

$$x^2 := \lambda_1^2 c_1 + \lambda_2^2 c_2 + \cdots + \lambda_k^2 c_k = x \circ x.$$ 

We also have the following definition.

**Definition 1.3.10.** Let $x$ be an element of a Euclidean Jordan algebra $J$ with a spectral decomposition $x := \lambda_1 c_1 + \lambda_2 c_2 + \cdots + \lambda_k c_k$. Then

1. the square root of $x$ is $x^{1/2} := \lambda_1^{1/2} c_1 + \lambda_2^{1/2} c_2 + \cdots + \lambda_k^{1/2} c_k$, whenever all $\lambda_i \geq 0$, and undefined otherwise;

2. the inverse of $x$ is $x^{-1} := \lambda_1^{-1} c_1 + \lambda_2^{-1} c_2 + \cdots + \lambda_k^{-1} c_k$, whenever all $\lambda_i \neq 0$, and undefined otherwise.
More generally, if $f$ is any real valued continuous function, then it is also possible to extend the above definition to define $f(x)$ as

$$f(x) := f(\lambda_1)c_1 + f(\lambda_2)c_2 + \cdots + f(\lambda_k)c_k.$$  

Observe that $x^{-1} \circ x = e$. We call $x$ invertible if $x^{-1}$ is defined, and non-invertible or singular otherwise. Note that every positive definite element is invertible and its inverse is also positive definite.

**Remark 1.** The equality $x \circ y = e$ may not imply that $y = x^{-1}$, as it can be seen in the following equality [18, Chapter II]:

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \circ \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$  

We define the differential operator $D_x : J \rightarrow J$ by

$$D_x f(x) := \left( \frac{d}{d\lambda_1} f(\lambda_1) \right) c_1 + \left( \frac{d}{d\lambda_2} f(\lambda_2) \right) c_2 + \cdots + \left( \frac{d}{d\lambda_k} f(\lambda_k) \right) c_k.$$  

The Jacobian matrix $\nabla_x f(x)$ is defined so that $(\nabla_x f(x))^T y = (D_x f(x)) \circ y$ for all $y \in J$. For $n \geq 2$, we define $D^n_x f(x)$ recursively by $D^n_x f(x) := D_x(D^{n-1}_x f(x))$. For instance, $D^2_x x = D_x(D_x x) = D_x e = 0$, $D_x x^{-1} = -x^{-2}$, provided that $x$ is invertible.

More generally, if $y$ is a function of $x$ and they are both simultaneously decomposed, then $D_x f(y) := D_y f(y) \circ D_x y$. For example, $D_x y^{-1} = -y^{-2} \circ D_x y$, provided that $y$ is invertible. Note that, if $y$ and $z$ are functions of $x$ and they are all simultaneously decomposed, then $D_x(y \pm z) = D_x y \pm D_x z$, and $D_x(y \circ z) = (D_x y) \circ z + y \circ (D_x z)$.

This differential operator is interesting and will play an important role when computing partial derivatives.
There is a one-to-one correspondence between Euclidean Jordan algebras and symmetric cones.

**Definition 1.3.11.** If \( J \) is a Euclidean Jordan algebra then its *cone of squares* is the set

\[
\mathcal{K}_J := \{ x^2 : x \in J \}.
\]

Such a one to one correspondence between (cones of squares of) Euclidean Jordan algebras and symmetric cones is given by the following fundamental result, which says that a cone is symmetric if and only if it is the cone of squares of some Euclidean Jordan algebra.

**Theorem 1.3.3** (Jordan algebraic characterization of symmetric cones, [18]). A regular cone \( K \) is symmetric iff \( K = \mathcal{K}_J \) for some Euclidean Jordan algebra \( J \).

The above result implies that an element is positive semidefinite if and only if it belongs to the cone of squares, and it is positive definite if and only if it belongs to the interior of the cone of squares. In other words, an element \( x \) in a Euclidean Jordan algebra \( J \) is positive semidefinite if and only if \( x \in \mathcal{K}_J \), and is positive definite if and only if \( x \in \text{int}(\mathcal{K}_J) \), where \( \text{int}(\mathcal{K}_J) \) denotes the interior of the cone \( \mathcal{K}_J \).

The following notations will be used throughout the dissertation. For a Euclidean Jordan algebra \( J \), we write \( x \succeq_{\mathcal{K}_J} 0 \) and \( x \succ_{\mathcal{K}_J} 0 \) to mean that \( x \in \mathcal{K}_J \) and \( x \in \text{int}(\mathcal{K}_J) \), respectively. We also write \( x \succeq_{\mathcal{K}_J} y \) (or \( y \preceq_{\mathcal{K}_J} x \)) and \( x \succ_{\mathcal{K}_J} y \) (or \( y \prec_{\mathcal{K}_J} x \)) to mean that \( x - y \succeq_{\mathcal{K}_J} 0 \) and \( x - y \succ_{\mathcal{K}_J} 0 \), respectively.

**Example 19.** The cone \( \mathcal{K}_{S_n} \) is, indeed, \( S^n_+ \), the cone of real symmetric positive semidefinite matrices of order \( n \). This can be seen in view of the fact that a symmetric matrix is square of another symmetric matrix if and only if it is a positive semidefinite matrix. To prove this fact, suppose that \( X \) is positive semidefinite, then it has nonnegative eigenvalues...
\(\lambda_1, \lambda_2, \ldots, \lambda_n\) and, by the spectral theorem for real symmetric matrices, there exists an orthogonal matrix \(Q \in \mathbb{R}^{n \times n}\) and a diagonal matrix \(\Lambda := \text{diag}(\lambda_1; \lambda_2; \ldots; \lambda_n) \in \mathbb{R}^{n \times n}\) such that \(X = Q\Lambda Q^T\). By letting \(Y := Q\Lambda^{1/2}Q^T \in \mathcal{S}^n\) where \(\Lambda^{1/2} := \text{diag}(\sqrt{\lambda_1}; \sqrt{\lambda_2}; \ldots; \sqrt{\lambda_n})\), it follows that

\[
X = Q\Lambda Q^T = Q\Lambda^{1/2}\Lambda^{1/2}Q^T = (Q\Lambda^{1/2}Q^T)(Q\Lambda^{1/2}Q^T) = Y^2.
\]

To prove the other direction, let us assume that \(X = Y^2\) for some \(Y \in \mathcal{S}^n\). It is clear that \(X \in \mathcal{S}^n\). To show that \(X\) is positive semidefinite, let \((\lambda, \mathbf{v})\) be an eigenpair of \(Y\), then \(\lambda \in \mathbb{R}\) (every real symmetric matrix has real eigenvalues). Furthermore, we have that \(X\mathbf{v} = Y^2\mathbf{v} = Y(\lambda\mathbf{v}) = \lambda^2\mathbf{v}\). Thus, \((\lambda^2, \mathbf{v})\) is an eigenpair of \(X\), which means that \(X\) has always nonnegative eigenvalues, and therefore this completes the proof.

**Example 20.** The cone of squares of \(\mathcal{E}^n\) is the second-order cone of dimension \(n\):

\[
\mathcal{E}^n_+ := \{\xi \in \mathcal{E}^n : \xi_0 \geq \|\xi\|_2\}.
\]

To see this, recall that the cone of squares (with with respect to “\(\circ\)” defined in Example 4) is

\[
\mathcal{K}_{\mathcal{E}^n} = \{\zeta^2 : \zeta \in \mathcal{E}^n\} = \{(\|\zeta\|_2^2 ; 2\zeta_0\zeta) : \zeta \in \mathcal{E}^n\}.
\]

Thus, any \(x \in \mathcal{K}_{\mathcal{E}^n}\) can be written as \(x = (\|y\|_2^2 ; 2y_0\bar{y})\), for some \(y \in \mathcal{E}^n\). It follows that

\[
x_0 = \|y\|_2^2 = y_0^2 + \|\bar{y}\|_2^2 \geq 2y_0\bar{y} = \bar{x},
\]

where the inequality follows by observing that \((y_0 - \|\bar{y}\|_2)^2 \geq 0\). This means that \(x \in \mathcal{E}^n_+\) and hence \(\mathcal{K}_{\mathcal{E}^n} \subseteq \mathcal{E}^n_+\). The proof of the other direction can be found in \(\S 4\) of [1].

---

2 We indicate by the operator \(\text{diag}(\cdot)\) that one maps its argument to a block diagonal matrix; for example, if \(x \in \mathbb{R}^n\), then \(\text{diag}(x)\) is the \(n \times n\) diagonal matrix with the entries of \(x\) on the diagonal.
Theorem 1.3.4 ([18, Theorem III.1.5]). Let \( \mathcal{J} \) be a Jordan algebra over \( \mathbb{R} \) with the identity element \( e \). The following properties are equivalent:

1. \( \mathcal{J} \) is a Euclidean Jordan algebra.

2. The symmetric bilinear form \( \text{trace}(x \circ y) \) is positive definite.

A direct consequence of the above theorem is that if \( \mathcal{J} \) is a Euclidean Jordan algebra, then \( x \cdot y := \text{trace}(x \circ y) \) is an inner product. In the sequel, we define the inner product as \( (x, y) := x \cdot y = \text{trace}(x \circ y) \) and we call it the Frobenius inner product. It is easy to see that, for \( x, y, z \in \mathcal{J} \), \( x \cdot e = \text{trace}(x) \), \( x \cdot y = y \cdot x \), \( (x + y) \cdot z = x \cdot z + y \cdot z \), \( x \cdot (y + z) = x \cdot z + x \cdot z \), and \( (x \circ y) \cdot z = x \cdot (y \circ z) \).

For \( x \in \mathcal{J} \), the Frobenius norm (denoted by \( \| \cdot \|_F \), or simply \( \| \cdot \| \)) is defined as \( \| x \| := \sqrt{x \cdot x} \). We can also define various norms on \( \mathcal{J} \) as functions of eigenvalues. For example, the definition of the Frobenius norm can be rewritten as \( \| x \| := \sqrt{\lambda_1^2 + \lambda_2^2 + \cdots + \lambda_k^2} = \sqrt{\text{trace}(x^2)} = \sqrt{x \cdot x} \). The Cauchy-Schwartz inequality holds for the Frobenius inner product, i.e., for \( x, y \in \mathcal{J} \), \( |x \cdot y| \leq \| x \| \| y \| \) (see for example [34]). Note that \( \| e \| = \sqrt{e \cdot e} = \sqrt{r} \).

Let \( x \in \mathcal{J} \) and \( t \in \mathbb{R} \). For a function \( g := g(x, t) \) from \( \mathcal{J} \times \mathbb{R} \) into \( \mathbb{R} \), we will use “\( g \)” for the partial derivative of \( g \) with respect to \( t \), and “\( \nabla_x g \)”, “\( \nabla^2_{xx} g \)”, “\( \nabla^3_{xxx} g \)” to denote the gradient, Hessian, and the third order partial derivative of \( g \) with respect to \( x \). For a function \( y := y(x, t) \) from \( \mathcal{J} \times \mathbb{R} \) into \( \mathcal{J} \), we will also use “\( y \)” for the partial derivative of \( y \) with respect to \( t \), “\( D_x y \)” to denote the first partial derivative of \( y \) with respect to \( x \), and “\( \nabla_x y \)” to denote the Jacobian matrix of \( y \) (with respect to \( x \)).

Let \( h_1, h_2, \ldots, h_k \in \mathcal{J} \). For a function \( f \) from \( \mathcal{J} \) into \( \mathbb{R} \), we write \( \nabla^k x \ldots x f(x)[h_1, h_2, \ldots, h_k] := \frac{\partial^k f(x + t_1 h_1 + \cdots + t_k h_k)}{\partial t_1 \cdots \partial t_k}|_{t_1 = \cdots = t_k = 0} \) to denote the value of the \( k \)th differential of \( f \) taken at \( x \) along the directions \( h_1, h_2, \ldots, h_k \).
We now present some handy tools that will help with our computations.

**Lemma 1.3.1.** Let $J$ be a Euclidean Jordan algebra with identity $e$, and $x, y, z \in J$. Then

1. $\{\ln \det (e + tx)\}'|_{t=0} = \text{trace}(x)$ and, more generally, $\{\ln \det (y + tx)\}'|_{t=0} = y^{-1} \bullet x$, provided that $\det (e + tx)$ and $\det (y + tx)$ are positive.

2. $\{\text{trace}(e + tx)^{-1}\}'|_{t=0} = -\text{trace}(x)$ and, more generally, $\{\text{trace}(e + tx \circ y)^{-1}\}'|_{t=0} = -x \bullet y$, provided that $e + tx$ is invertible.

3. $\nabla_x \ln \det x = x^{-1}$, provided that $\det x$ is positive (so $x$ is invertible). More generally, if $y$ is a function of $x$, then $\nabla_x \ln \det y = (\nabla_x y)^\top y^{-1}$, provided that $\det y$ is positive.

4. $\nabla_x x^{-1} = -[x^{-1}]$ provided that $x$ is invertible, and hence $\nabla^2_{xx} \ln \det x = \nabla_x x^{-1} = -[x^{-1}]$. More generally, if $y$ is a function of $x$, then $\nabla_x y^{-1} = -[y^{-1}] \nabla_x y$ provided that $y$ is invertible.

5. $\nabla_x [x][y] = 2[x, y]$. 

6. If $x$ and $y$ are functions of $t$ where $t \in \mathbb{R}$, then $(x \circ y)' = x \circ y' + x' \circ y$. In other words, $([x][y])' = [x'][y] + [x][y']$. Therefore, $[x]' = [x']$, $[x, y]' = [x, y'] + [x', y]$, and, in particular, $[x]' = 2[x, x']$.

7. $\nabla_x \text{trace}(x) = e = D_x x$ and, more generally, if $y$ is a function of $x$ and they are both simultaneously decomposed, then $\nabla_x \text{trace}(y) = (\nabla_x y)^\top e = D_x y$. Hence, if $y$ and $z$ are functions of $x$ and they are all simultaneously decomposed, then

$$\nabla_x (y \bullet z) = D_x (y \circ z) = (D_x y) \circ z + (D_x z) \circ y = (\nabla_x y)^\top z + (\nabla_x z)^\top y.$$
The proofs of most of these statements are straightforward. We only indicate that item 1 follows from the facts that \( \{ \det (e + tx) \}'|_{t=0} = \text{trace}(x) \) and that \( \{ \det (y + tx) \}'|_{t=0} = \det(y)(y^{-1} \bullet x) \) [18, Proposition III.4.2], the first statement in item 2 follows from spectral decomposition (I), the first statement in item 3 is taken from [18, Proposition III.4.2], the first statement in item 4 is taken from [18, Proposition II.3.3], item 5 is taken from §3 of [18, Chapter II], the first statement in item 6 is taken from §4 of [18, Chapter II], and the first statement in item 7 is obtained by using item 3 and the observation that \( \det(\exp(x)) = \exp(\text{trace}(x)) \).

**Definition 1.3.12.** We say two elements \( x, y \) of a Euclidean Jordan algebra \( J \) operator commute if
\[
\lfloor x \rfloor \lfloor y \rfloor = \lfloor y \rfloor \lfloor x \rfloor.
\]
In other words, \( x \) and \( y \) operator commute if for all \( z \in J \), we have that \( x \circ (y \circ z) = y \circ (x \circ z) \).

We remind the reader about the fact that two matrices \( X, Y \in \mathcal{S}^n \) commute if and only if \( XY \) is symmetric, if and only if \( X \) and \( Y \) can be simultaneously diagonalized, i.e. they share a common system of orthonormal eigenvectors (the same \( Q \)). This fact can be generalized in the following theorem which can be also applied to multiple-block elements.

**Theorem 1.3.5 ([36, Theorem 27]).** Two elements of a Euclidean Jordan algebra operator commute if and only if they are simultaneously decomposed.

Note that if two operator commutative elements \( x \) and \( y \) are invertible, then so is \( x \circ y \). Moreover, \( (x \circ y)^{-1} = x^{-1} \circ y^{-1} \), and \( \det(x \circ y) = \det(x) \det(y) \) (see also [18, Proposition II.2.2]). In Remark 1 we mentioned that the equality \( x \circ y = e \) may not imply that \( y = x^{-1} \). However, the equality \( x \circ y = e \) does imply that \( y = x^{-1} \) when the elements \( x \) and \( y \) operator commute.

**Lemma 1.3.2 (Properties of \( \lceil \cdot \rceil \)).** Let \( x \) and \( y \) be elements of a Euclidean Jordan algebra with rank \( r \) and dimension \( n \), \( x \) invertible, and \( k \) be an integer. Then
1. \( \det(\lceil x \rceil y) = \det^2(x) \det(y) \).
2. $[x]x^{-1} = x, [x]e = x^2$.

3. $[[y]x] = [y][x][y]$.

4. $[x^k] = [x]^k$.

The first three items of the following lemma are taken from [18, Chapters II and III] and the last one is taken from [36, §2].

We use "," for adjoining elements of a Euclidean Jordan algebra $J$ in a row, and use ";" for adjoining them in a column. Thus, if $J$ is a Euclidean Jordan algebra, and $x_i \in J$ for $i = 1, 2, \ldots, m$, we have

$$(x_1; x_2; \ldots; x_m) := \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \in J \times J \times \cdots \times J, \quad \text{m times}$$

and we write $(x_1; x_2; \ldots; x_m)^T := (x_1, x_2, \ldots, x_m) := [x_1 \ x_2 \ \cdots \ x_m]$. As we mentioned earlier, we also use the superscript "T" to indicate transposition of column vectors in $\mathbb{R}^n$.

We end this section with the following lemma which is essentially a part of Lemma 12 in [36].

**Lemma 1.3.3.** Let $x \in J$ with a spectral decomposition $x = \lambda_1 c_1 + \lambda_2 c_2 + \cdots + \lambda_r c_r$. Then the following statements hold.

1. The matrices $[x]$ and $[\bar{x}]$ commute and thus share a common system of eigenvectors.

2. Every eigenvalue of $[x]$ have the form $(1/2)(\lambda_i + \lambda_j)$ for some $i, j \leq r$. In particular, $x \succeq_s 0 \ (x \succ_s 0)$ if and only if $[x] \succeq 0 \ ([x] \succ 0)$. The eigenvalues of $x$; $\lambda_i$’s, are amongst the eigenvalues of $[x]$. 

27
Chapter 2

Stochastic Symmetric Optimization Problems

In this chapter, we use the Jordan algebraic characterization of symmetric cones to define the SSP problem in both the primal and dual standard forms. We then see how this problem can include some general optimization problems as special cases.

2.1 The stochastic symmetric optimization problem

We define a problem based on the DSP problem analogous to the way SLP problem is defined based on the DLP problem. To do so, we first introduce the definition of a DSP.

Let \( \mathcal{J} \) be a Euclidean Jordan algebra with dimension \( n \) and rank \( r \). The DSP problem and its dual [36] are

\[
\begin{align*}
\min \quad & c \cdot x \\
\text{s.t.} \quad & a_i \cdot x = b_i, i = 1, 2, \ldots, m \\
& x \succeq_{\mathcal{K}_J} 0;
\end{align*}
\]

\[
\begin{align*}
\max \quad & b^T y \\
\text{s.t.} \quad & \sum_{i=1}^{m} y_i a_i \preceq_{\mathcal{K}_J} c \\
& y \in \mathbb{R}^m,
\end{align*}
\]
where \( c, a_i \in J \) for \( i = 1, 2, \ldots, m \), \( b \in \mathbb{R}^m \), \( x \) is the primal variable, \( y \) is the dual variable, and, as mentioned in §2.1, \( K_J \) is the cone of squares of \( J \).

The pair \((P_1, D_1)\) can be rewritten in the following compact form

\[
\begin{align*}
\min \quad & c \cdot x \\
\text{s.t.} \quad & Ax = b \\
\end{align*}
\]

\[
\begin{align*}
\max \quad & b^T y \\
\text{s.t.} \quad & A^T y \preceq_{K_J} c \\
\end{align*}
\]

\[
\begin{align*}
\quad & x \succeq_{K_J} 0; \\
\quad & y \in \mathbb{R}^m,
\end{align*}
\]

where \( A := (a_1; a_2; \ldots; a_m) \) is a linear operator that maps \( J \) into \( \mathbb{R}^m \) and \( A^T \) is a linear operator that maps \( \mathbb{R}^m \) into \( J \) such that \( x \circ A^T y = (Ax)^T y \). In fact, we can prove weak and strong duality properties for the pair \((P_2, D_2)\) as justification for referring to them as a primal dual pair; see, for example, [30].

In the rest of this section, we assume that \( m_1, m_2, n_1, n_2, r_1, \) and \( r_2 \) are positive integers, and \( J_1 \) and \( J_2 \) are Euclidean Jordan algebras with identities \( e_1 \) and \( e_2 \), dimensions \( n_1 \) and \( n_2 \), and ranks \( r_1 \) and \( r_2 \), respectively.

### 2.1.1 Definition of an SSP in primal standard form

We define the primal form of the SSP based on the primal form of the DSP. Given \( a_i, t_j \in J_1 \) and \( w_j \in J_2 \) for \( i = 1, 2, \ldots, m_1 \) and \( j = 1, 2, \ldots, m_2 \). Let \( A := (a_1; a_2; \ldots; a_m) \) be a linear operator that maps \( x \) to the \( m_1 \)-dimensional vector whose \( i^{th} \) component is \( a_i \cdot x \), \( b \in \mathbb{R}^{m_1} \), \( c \in J_1 \), \( T := (t_1; t_2; \ldots; t_m) \) be a linear operator that maps \( x \) to the \( m_2 \)-dimensional vector whose \( i^{th} \) component is \( t_i \cdot x \), \( W := (w_1; w_2; \ldots; w_m) \) be a linear operator that maps \( y \) to the \( m_2 \)-dimensional vector whose \( i^{th} \) component is \( w_i \cdot y \), \( h \in \mathbb{R}^{m_2} \), and \( d \in J_2 \). We also assume that \( A, b \) and \( c \) are deterministic data, and \( T, W, h \) and \( d \) are random data whose realizations depend on an underlying outcome \( \omega \) in an event space \( \Omega \) with a known probability function \( P \). Given this data, an SSP with recourse in
primal standard form is
\[
\begin{align*}
\min \quad & c \cdot x + \mathbb{E} \left[ Q(x, \omega) \right] \\
\text{s.t.} \quad & Ax = b \\
& x \succeq_{\mathcal{K}, \mathcal{J}_1} 0,
\end{align*}
\] (2.1.1)
where \( Q(x, \omega) \) is the minimum value of the problem
\[
\min \quad d(\omega) \cdot y \\
\text{s.t.} \quad W(\omega)y = h(\omega) - T(\omega)x \\
& y \succeq_{\mathcal{K}, \mathcal{J}_2} 0,
\] (2.1.2)
where \( x \) is the first-stage decision variable, \( y \) is the second-stage variable, and
\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).
\]

2.1.2 Definition of an SSP in dual standard form

In many applications we find that a defined SSP problem based on the dual standard form \((D_2)\) is more useful. In this part we define the dual form of the SSP based on the dual form of the DSP. Given \( a_i \in \mathcal{J}_1 \) and \( t_i, w_j \in \mathcal{J}_2 \) for \( i = 1, 2, \ldots, m_1 \) and \( j = 1, 2, \ldots, m_2 \). Let \( A := (a_1, a_2, \ldots, a_{m_1}), b \in \mathcal{J}_1, c \in \mathbb{R}^{m_1}, T := (t_1, t_2, \ldots, t_{m_1}), W := (w_1, w_2, \ldots, w_{m_2}), h \in \mathcal{J}_2, \) and \( d \in \mathbb{R}^{m_2} \). We also assume that \( A, b \) and \( c \) are deterministic data, and \( T, W, h \) and \( d \) are random data whose realizations depend on an underlying outcome \( \omega \) in an event space \( \Omega \) with a known probability function \( P \). Given this data, an SSP with recourse in dual standard form is
\[
\begin{align*}
\max \quad & c^T x + \mathbb{E} \left[ Q(x, \omega) \right] \\
\text{s.t.} \quad & Ax \preceq_{\mathcal{K}, \mathcal{J}_1} b,
\end{align*}
\] (2.1.3)
where \( Q(x, \omega) \) is the maximum value of the problem
\[
\begin{align*}
\max & \quad d(\omega)^T y \\
\text{s.t.} & \quad W(\omega)y \preceq_{K_{\rho_2}} h(\omega) - T(\omega)x,
\end{align*}
\] (2.1.4)

where \(x \in \mathbb{R}^{m_1}\) is the first-stage decision variable, \(y \in \mathbb{R}^{m_2}\) is the second-stage variable, and

\[E[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).\]

In fact, it is also possible to define SSPs in “mixed forms” where the first-stage is based on primal problem \((P_2)\) while the second-stage is based on the dual problem \((D_2)\), and vice versa.

### 2.2 Problems that can be cast as SSPs

It is interesting to mention that almost all conic optimization problems in real world applications are associated with symmetric cones, and that, as an illustration of the modeling power of conic programming, all deterministic convex programming problems can be formulated as deterministic conic programs (see [29]). As a consequence, by considering the stochastic counterpart of this result, it is straightforward to show that all stochastic convex programming problems can be formulated as stochastic conic programs.

In this section we will see how SSPs can include some general optimization problems as special cases. We start with two-stage stochastic linear programs with recourse.

**Problem 1. Stochastic linear programs:**

It is clear that the space of real numbers \(\mathbb{R}\) with Jordan multiplication \(x \circ y := xy\) and inner product \(x \bullet y := xy\) forms a Euclidean Jordan algebra. Since a real number is a square of another real number if and only if it is nonnegative, the cone of squares of \(\mathbb{R}\) is indeed \(\mathbb{R}_+\); the set of all nonnegative real numbers. This verifies that \(\mathbb{R}_+\) is a symmetric cone. The cone \(\mathbb{R}_+^p\) of nonnegative orthants of \(\mathbb{R}^p\) is also symmetric because it is just
Euclidean Jordan algebra \[ \mathbb{R}^p = \{(x_1, x_2, \ldots, x_p) : x_i \in \mathbb{R}, i = 1, 2, \ldots, p\} \]

Symmetric cone \[ \mathbb{R}^p_+ = \{x \in \mathbb{R}^p : x_i \geq 0, i = 1, 2, \ldots, p\} \]

Conic inequality \[ x \succeq \mathbb{R}^p_+ \equiv x \geq 0 \text{ and } x \succ \mathbb{R}^p_+ \equiv x > 0 \]

Jordan multiplication \[ x \circ y = (x_1 y_1, x_2 y_2, \ldots, x_p y_p) = \text{diag}(x) y \]

Inner product \[ x \bullet y = x_1 y_1 + x_2 y_2 + \cdots + x_p y_p = x^T y \]

Identity element \[ e = (1; 1; \ldots; 1) \]

Spectral decomposition \[ x = \sum_{i=1}^{p} x_i (1; 0; \ldots; 0) + \sum_{i=1}^{p} x_i (0; 1; \ldots; 0) + \sum_{i=1}^{p} x_i (0; 0; \ldots; 1) \]

Cone rank \[ \text{rank}(\mathbb{R}^p_+) = p \]

Expression for trace \[ \text{trace}(x) = x_1 + x_2 + \cdots + x_p \]

Expression for determinant \[ \text{det}(x) = x_1 x_2 \cdots x_p \]

Expression for Frobenius norm \[ ||x|| = \sqrt{x_1^2 + x_2^2 + \cdots + x_p^2} = ||x||_2 \]

Expression for inverse \[ x^{-1} = \left( \frac{1}{x_1}, \frac{1}{x_2}, \ldots, \frac{1}{x_p} \right) \text{ (if } x_i \neq 0 \text{ for all } i) \]

Expression for square root \[ x^{1/2} = (\sqrt{x_1}, \sqrt{x_2}, \ldots, \sqrt{x_p}) \text{ (if } x_i \geq 0 \text{ for all } i) \]

Log barrier function \[ -\ln \text{det}(x) = -\sum_{i=1}^{p} \ln x_i \text{ if } x_i > 0 \text{ for all } i \]

Table 2.1: The Euclidean Jordan algebraic structure of the nonnegative orthant cone.

When \( J_1 = \mathbb{R}^{n_1} \) and \( J_2 = \mathbb{R}^{n_2} \); the spaces of vectors of dimensions \( n_1 \) and \( n_2 \), respectively, with Jordan multiplication \( x \circ y := \text{diag}(x)y \) and the standard inner product \( x \bullet y := x^T y \), then \( \mathcal{K}_{J_1} = \mathbb{R}^{n_1}_+ \) and \( \mathcal{K}_{J_2} = \mathbb{R}^{n_2}_+ \); the cones of nonnegative orthants of \( \mathbb{R}^{n_1} \) and \( \mathbb{R}^{n_2} \), respectively, and hence the SSP problem (2.1.1, 2.1.2) becomes the SLP problem:

\[
\begin{align*}
\min \quad & c^T x + \mathbb{E} [Q(x, \omega)] \\
\text{s.t.} \quad & A x = b \\
& x \geq 0,
\end{align*}
\]
where \( Q(x, \omega) \) is the minimum value of the problem
\[
\begin{align*}
\min & \quad d(\omega)^T y \\
\text{s.t.} & \quad W(\omega)y = h(\omega) - T(\omega)x \\
& \quad y \geq 0.
\end{align*}
\]

Two-stage SLP with recourse has many practical applications, see for example [15].

**Problem 2. Stochastic second-order cone programs:**

When \( J_1 = \mathcal{E}^{n_1} \) and \( J_2 = \mathcal{E}^{n_2} \) with Jordan multiplication \( x \circ y := (x^T y, x_0 y + y_0 x) \), and inner product \( x \bullet y := x^T y \), then \( K_{J_1} = \mathcal{E}^{n_1}_+ \) and \( K_{J_2} = \mathcal{E}^{n_2}_+ \); the second-order cones of dimensions \( n_1 \) and \( n_2 \), respectively (see Example 20), and hence we obtain SSOCP with recourse. Table 2.2\(^1\) summarizes the Euclidean Jordan algebraic structure of the second-order cone in both single- and multiple-block forms.

To introduce the SSOCP problem, we first introduce some notations that will be used throughout this part and in Chapter 5.

For simplicity sake, we write the *single-block* second-order cone inequality \( x \succeq_{\mathcal{E}_+^p} 0 \) as \( x \succeq 0 \) (to mean that \( x \in \mathcal{E}_+^p \)) when \( p \) is known from the context, and the *multiple-block* second-order cone inequality \( x \succeq_{\mathcal{E}_+^{p_1} \times \mathcal{E}_+^{p_2} \times \cdots \times \mathcal{E}_+^{p_N}} 0 \) as \( x \succeq_N 0 \) (to mean that \( x \in \mathcal{E}_+^{p_1} \times \mathcal{E}_+^{p_2} \times \cdots \times \mathcal{E}_+^{p_N} \)) when \( p_1, p_2, \ldots, p_N \) are known from the context. It is immediately seen that, for every vector \( x \in \mathbb{R}^p \) where \( p = \sum_{i=1}^{N} p_i \), \( x \succeq_N 0 \) if and only if \( x \) is partitioned conformally as \( x = (x_1; x_2; \ldots; x_N) \) and \( x_i \succeq 0 \) for \( i = 1, 2, \ldots, N \). We also write \( x \succeq y \ (x \succeq_N y) \) or \( y \succeq x \ (y \succeq_N x) \) to mean that \( x - y \succeq 0 \ (x - y \succeq_N 0) \).

We now are ready to introduce the definition of an SSOCP with recourse.

Let \( N_1, N_2 \geq 1 \) be integers. For \( i = 1, 2, \ldots, N_1 \) and \( j = 1, 2, \ldots, N_2 \), let \( m_1, m_2, n_1, n_2, n_{1i}, n_{2j} \) be positive integers such that \( n_1 = \sum_{i=1}^{N_1} n_{1i} \) and \( n_2 = \sum_{i=1}^{N_2} n_{2j} \). An SSOCP with recourse in primal standard form is defined based on deterministic data \( A \in \mathbb{R}^{m_1 \times n_1}, b \in \mathbb{R}^{m_1}, C \in \mathbb{R}^{m_2 \times n_2}, d \in \mathbb{R}^{m_2} \) as
\[
\begin{align*}
\min & \quad d(\omega)^T y \\
\text{s.t.} & \quad W(\omega)y = h(\omega) - T(\omega)x \\
& \quad y \geq 0,
\end{align*}
\]

\(^1\)The direct sum of two square matrices \( A \) and \( B \) is the block diagonal matrix \( A \oplus B := \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \).
Table 2.2: The Euclidean Jordan algebraic structure of the second-order cone in single- and multiple-block forms.

$\mathbb{R}^{m_1}$ and $c \in \mathbb{R}^{n_1}$ and random data $T \in \mathbb{R}^{m_2 \times n_1}, W \in \mathbb{R}^{m_2 \times n_2}, h \in \mathbb{R}^{m_2}$ and $d \in \mathbb{R}^{n_2}$ whose realizations depend on an underlying outcome $\omega$ in an event space $\Omega$ with a known probability function $P$. Given this data, the two-stage SSOCP in the primal standard form is the problem

$$
\min \ c^T x + \mathbb{E} \ [Q(x, \omega)]
$$

s.t. \quad Ax = b \quad (2.2.1)

$$
x \succeq_{N_1} 0,
$$

where $x \in \mathbb{R}^{n_1}$ is the first-stage decision variable and $Q(x, \omega)$ is the minimum value of the problem

$$
\min \ d(\omega)^T y
$$

s.t. \quad W(\omega) y = h(\omega) - T(w)x \quad (2.2.2)

$$
y \succeq_{N_2} 0,
$$

<table>
<thead>
<tr>
<th>Euclidean Jordan alg. $\mathcal{J}$</th>
<th>$\mathcal{E}^P = {x : x = (x_0; \bar{x}) \in \mathbb{R} \times \mathbb{R}^{P-1}}$</th>
<th>$\mathcal{E}^{P_1} \times \cdots \times \mathcal{E}^{P_N}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symmetric cone $\mathcal{K}_J$</td>
<td>$\mathcal{E}_J^P = {x \in \mathcal{E}^P : x_0 \geq |\bar{x}|_2}$</td>
<td>$\mathcal{E}_J^{P_1} \times \cdots \times \mathcal{E}_J^{P_N}$</td>
</tr>
<tr>
<td>$x \succeq_{\mathcal{K}<em>J} 0$ $(x \succ</em>{\mathcal{K}_J} 0)$</td>
<td>$x \geq 0$ $(x &gt; 0)$</td>
<td>$x \succeq_{\mathcal{K}^F} 0$ $(x \succ_{\mathcal{K}^F} 0)$</td>
</tr>
<tr>
<td>Jordan product $x \circ y$</td>
<td>$(x^1 y; x_0 y + y_0 \bar{x}) = \text{Arw}(x)y$</td>
<td>$(x_1 \circ y_1; \ldots; x_N \circ y_N)$</td>
</tr>
<tr>
<td>Inner product $x \bullet y$</td>
<td>$x^1 y$</td>
<td>$x_1 y_1 + \cdots + x_N y_N$</td>
</tr>
<tr>
<td>The identity $e$</td>
<td>$(1; 0)$</td>
<td>$(e_1; \ldots; e_N)$</td>
</tr>
</tbody>
</table>

The matrix $[x]$ $\text{Arw}(x)$ $\text{Arw}(x_1) \oplus \cdots \oplus \text{Arw}(x_N)$

Spectral decomposition $x = \left(\begin{array}{c} x_0 + \|\bar{x}\|_2 \left(\frac{1}{2}\right) \left(\begin{array}{c} \lambda_1 \c_1 \\ \lambda_2 \c_2 \end{array}\right) \end{array}\right)$ $x_0 - \|\bar{x}\|_2 \left(\frac{1}{2}\right) \left(\begin{array}{c} 1 - \frac{x}{\|\bar{x}\|_2} \\ 1 - \frac{-x}{\|\bar{x}\|_2} \end{array}\right)$

$\text{rank}(\mathcal{K}_J)$ $2$ $2N$

Expression for trace($x$) $\lambda_1 + \lambda_2 = 2x_0$ $\sum_{i=1}^N \text{trace}(x_i)$

Expression for det($x$) $\lambda_1 \lambda_2 = \lambda_0^2 - \|\bar{x}\|^2$ $\prod_{i=1}^N \text{det}(x_i)$

Frobenius norm $\|x\|_F$ $\sqrt{\lambda_1^2 + \lambda_2^2} = \sqrt{2} \|\bar{x}\|_2$ $\sum_{i=1}^N \|x_i\|_F^2$

Inverse $x^{-1}$ $\lambda_1^{-1} c_1 + \lambda_2^{-1} c_2 = \frac{x}{\text{det}(x)}$ $(x_1^{-1}; \ldots; x_N^{-1})$ (if $\text{det}(x) \neq 0$; o/w $x$ is singular) \[(x_1^{-1}; \ldots; x_N^{-1}) \] (if $x_i^{-1}$ exists for all $i$; o/w $x$ is singular)

Log barrier function $-\ln \left(x_0^2 - \|\bar{x}\|_2^2\right)$ $-\sum_{i=1}^N \ln \text{det} x_i$
where \( y \in \mathbb{R}^{n_2} \) is the second-stage variable and

\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).
\]

Note that if \( N_1 = n_1 \) and \( N_2 = n_2 \), then SSOCP (2.2.1, 2.2.2) reduces to SLP. In fact, if \( N_1 = n_1 \), then \( n_i = 1 \) for each \( i = 1, 2, \ldots, N_1 \), and so \( x_i \in \mathcal{E}_+ := \{ t \in \mathbb{R} : t \geq 0 \} \) for each \( i = 1, 2, \ldots, N_1 \). Thus, the constraint \( x \succeq_{N_1} 0 \) means the same as \( x \geq 0 \), i.e., \( x \) lies in the nonnegative orthant of \( \mathbb{R}^{n_1} \). The same situation occurs for \( y \succeq N_2 \). Thus SLPs is a special case of SSOCP (2.2.1, 2.2.2).

Stochastic quadratic programs (SQPs) are also a special case of SSOCPS. To demonstrate this, recall that a two-stage SMIQP with recourse is defined based on deterministic data \( C \in S_+^{n_1}, c \in \mathbb{R}^{n_1}, A \in \mathbb{R}^{m_1 \times n_1} \) and \( b \in \mathbb{R}^{m_1} \); and random data \( H \in S_+^{n_2}, d \in \mathbb{R}^{n_2}, T \in \mathbb{R}^{m_2 \times n_1}, W \in \mathbb{R}^{m_2 \times n_2} \), and \( h \in \mathbb{R}^{m_2} \) whose realizations depend on an underlying outcome in an event space \( \Omega \) with a known probability function \( P \). Given this data, an SQP with recourse is

\[
\begin{align*}
\min_{x, \omega} \quad & q_1(x, \omega) = x^T C x + c^T x + \mathbb{E}[Q(x, \omega)] \\
\text{s.t.} \quad & Ax = b \\
& x \succeq 0,
\end{align*}
\]

(2.2.3)

where \( x \in \mathbb{R}^{n_1} \) is the first-stage decision variable and \( Q(x, \omega) \) is the minimum value of the problem

\[
\begin{align*}
\min_{y, \omega} \quad & q_2(y, \omega) = y^T H(\omega) y + d(\omega)^T y \\
\text{s.t.} \quad & W(\omega) y = h(\omega) - T(\omega) x \\
& y \succeq 0,
\end{align*}
\]

(2.2.4)

where \( y \in \mathbb{R}^{n_2} \) is the second-stage decision variable and

\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).
\]
Observe that the objective function of (2.2.3) can be written as (see [1])

\[ q_1(x_1, \omega) = ||\bar{u}||^2 + \mathbb{E}[Q(x, \omega)] - \frac{1}{4}c^T C^{-1}c \] where \( \bar{u} = C^{1/2}x + \frac{1}{2}C^{-1/2}c. \)

Similarly, the objective function of (2.2.4) can be written as

\[ q_2(y, \omega) = ||\bar{v}||^2 - \frac{1}{4}d(\omega)^T H(\omega)^{-1}d(\omega) \] where \( \bar{v} = H(\omega)^{1/2}y + \frac{1}{2}H(\omega)^{-1/2}d(\omega). \)

Thus, problem (2.2.3, 2.2.4) can be transformed into the SSOCp:

\[
\begin{align*}
\min \quad & u_0 \\
\text{s.t.} \quad & \bar{u} - C^{1/2}x = \frac{1}{2}C^{-1/2}c \\
& Ax = b \\
& u \succeq 0, \quad x \geq 0,
\end{align*}
\]

(2.2.5)

where \( Q(x, \omega) \) is the minimum value of the problem

\[
\begin{align*}
\min \quad & v_0 \\
\text{s.t.} \quad & \bar{v} - H(\omega)^{1/2}y = \frac{1}{2}H(\omega)^{-1/2}d(\omega) \\
& W(\omega)y = h(\omega) - T(\omega)x \\
& v \succeq 0, \quad y \geq 0,
\end{align*}
\]

(2.2.6)

where

\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega)P(d\omega).
\]

Note that the SQP problem (2.2.3, 2.2.4) and the SSOCp problem (2.2.5, 2.2.6) will have the same minimizing solutions, but their optimal objective values are equal up to constants. More precisely, the difference between the optimal objective values of (2.2.4, 2.2.6) would be \(-\frac{1}{2}d(\omega)^T H(\omega)^{-1}d(\omega).\) Similarly, the optimal objective values of (2.2.3,
2.2.4) and (2.2.5, 2.2.6) will differ by
\[ -\frac{1}{2} c^T C^{-1} c - \frac{1}{2} \int_\Omega (d(\omega)^T H(\omega)^{-1} d(\omega)) P(d\omega). \]

In §5.1 of Chapter 5, we describe two applications that illustrate the applicability of two-stage SSOCPs with recourse.

**Problem 3. Stochastic rotated quadratic cone programs:**

For each vector \( x \in \mathbb{R}^n \) indexed from 0, we write \( \hat{x} \) for the sub-vector consisting of entries 2 through \( n - 1 \); therefore \( x = (x_0; x_1; \hat{x}) \in \mathbb{R} \times \mathbb{R} \times \mathbb{R}^{n-2} \). We let \( \hat{E}^n \) denote the \( n \) dimensional real vector space \( \mathbb{R} \times \mathbb{R} \times \mathbb{R}^{n-2} \) whose elements \( x \) are indexed with 0.

The rotated quadratic cone \( [1] \) of dimension \( n \) is defined by \( \hat{E}^n_+ := \{ x = (x_0; x_1; \hat{x}) \in \hat{E}^n : 2x_0 x_1 \geq ||\hat{x}||^2, \ x_0 \geq 0, x_1 \geq 0 \} \).

The constraint on \( x \) that satisfies the relation \( 2x_0 x_1 \geq ||\hat{x}||^2 \) is called a hyperbolic constraint. It is clear that the cones \( E^+_n \) and \( \hat{E}^n_+ \) have the same Euclidean Jordan algebraic structure. In fact, the later is obtained by rotating the former through an angle of thirty degrees in the \( x_0 x_1 \)-plane \( [1] \). More specifically, by writing \( x \succcurlyeq 0 \) \( (x \succcurlyeq_N 0) \) to mean that \( x \in \hat{E}^+_n \) \( (x \in E^+_n \times E^+_n \times \cdots \times E^+_n) \), then one can easily see that the hyperbolic constraint \( (x_0; x_1; \hat{x}) \succcurlyeq 0 \) is equivalent to the second-order cone constraint \( (2x_0 + x_1; 2x_0 - x_1; 2\hat{x}) \succcurlyeq 0 \). In fact,

\[
2x_0 + x_1 \geq ||(2x_0 - x_1; 2\hat{x})||_2 \iff (2x_0 + x_1)^2 \geq ||(2x_0 - x_1; 2\hat{x})||_2^2 \\
\iff 4x_0 x_1 \geq -4x_0 x_1 + 4||\hat{x}||_2^2 \\
\iff 2x_0 x_1 \geq ||\hat{x}||_2^2.
\]

So, if we are given the same setting as in Problem (2.2.1, 2.2.2), the two-stage stochastic rotated quadratic cone program (SRQCP) in the primal standard form is the problem
min $c^T x + \mathbb{E} [Q(x, \omega)]$
\[ \text{s.t. } A x = b \]
\[ x \succeq_{N_1} 0, \]

where $x \in \mathbb{R}^{n_1}$ is the first-stage decision variable and $Q(x, \omega)$ is the minimum value of the problem
\[
\min \ d(\omega)^T y \\
\text{s.t. } W(\omega) y = h(\omega) - T(w) x \\
y \succeq_{N_2} 0.
\]

In §5.2 of Chapter 5, we describe two applications that illustrate the applicability of two-stage SRQCPs with recourse.

**Problem 4. Stochastic semidefinite programs:**

When $J_1 = \mathcal{S}^{n_1}$ and $J_2 = \mathcal{S}^{n_2}$; the spaces of real symmetric matrices of orders $n_1$ and $n_2$, respectively, with Jordan multiplication $X \circ Y := \frac{1}{2} (XY + YX)$ and inner product $X \bullet Y := \text{trace}(XY)$, then $K_{J_1} = \mathcal{S}_+^{n_1}$ and $K_{J_2} = \mathcal{S}_+^{n_1}$; the cones of real symmetric positive semidefinite matrices of orders $n_1$ and $n_2$, respectively (see Example 19). We simply write the linear matrix inequality $X \succeq_{S_+} 0$ as $X \succeq 0$ (to mean that $X \in \mathcal{S}_+^p$). Hence the SSP problem (2.1.1, 2.1.2) becomes the SSDP problem (see also [9, Subsection 2.1]):

\[
\begin{align*}
\min \ & C \bullet X + \mathbb{E} [Q(X, \omega)] \\
\text{s.t. } & AX = b \\
& X \succeq 0,
\end{align*}
\]  
(2.2.7)

where $Q(X, \omega)$ is the minimum value of the problem

\[
\begin{align*}
\min \ & D(\omega) \bullet Y \\
\text{s.t. } & W(\omega) Y = h(\omega) - T(\omega) X \\
& Y \succeq 0,
\end{align*}
\]  
(2.2.8)
| Euclidean Jordan alg. $\mathcal{J}$ | $\mathcal{S}^p = \{X \in \mathbb{R}^{p \times p} : X^T = X\}$ | $\text{vec}(\mathcal{S}^p) = \{\text{vec}(X) \in \mathbb{R}^{p^2} : X^T = X\}$ |
| Symmetric cone $\mathcal{K}_\mathcal{J}$ | $\mathcal{S}_+^p = \{X \in \mathcal{S}^p : X \succeq 0\}$ | $\text{vec}(\mathcal{S}_+^p) = \{\text{vec}(X) \in \mathbb{R}^{p^2} : X \succeq 0\}$ |
| Jordan product $x \circ y$ | $X \circ Y = \frac{1}{2}(XY + YX)$ | $\text{vec}(X) \circ \text{vec}(Y) = \frac{1}{2} \text{vec}(XY + YX)$ |
| Inner product $x \bullet y$ | $X \bullet Y = \text{trace}(XY)$ | $\text{vec}(X) \bullet \text{vec}(Y) = \text{vec}(X)^T \text{vec}(Y)$ |
| The identity $e$ | $I_p$ | $\text{vec}(I_p)$ |
| The matrix $[X]$ | $\frac{1}{2}(I \otimes X + X \otimes I)$ | $\frac{1}{2}(I \otimes X + X \otimes I)$ |
| The matrix $|X|$ | $X \otimes X$ | $X \otimes X$ |
| Spectral decomposition | $X = \lambda_1 q_1q_1^T + \lambda_2 q_2q_2^T + \cdots + \lambda_p q_pq_p^T \begin{bmatrix} c_1 \\ C_1 \\ c_2 \\ C_2 \\ \vdots \\ c_p \\ C_p \end{bmatrix}$ | $\text{vec}(X) = \lambda_1 \text{vec}(q_1q_1^T) + \lambda_2 \text{vec}(q_2q_2^T) + \cdots + \lambda_p \text{vec}(q_pq_p^T) \begin{bmatrix} c_1 \\ C_1 \\ c_2 \\ C_2 \\ \vdots \\ c_p \\ C_p \end{bmatrix}$ |
| rank($\mathcal{K}_\mathcal{J}$) | $p$ | $p$ |

Table 2.3: The Euclidean Jordan algebraic structure of the positive semidefinite cone in matrix and vectorized matrix forms.

where $X \in \mathcal{S}^{n_1}$ is the first-stage decision variable, $Y \in \mathcal{S}^{n_2}$ is the second-stage variable, and

$$\mathbb{E}[Q(X,\omega)] := \int_{\Omega} Q(X,\omega)P(d\omega).$$

Here, clearly, $C \in \mathcal{S}^{n_1}$, $D \in \mathcal{S}^{n_2}$, and the linear operator $A : \mathcal{S}^{n_1} \longrightarrow \mathbb{R}^{m_1}$ is defined by

$$AX := (A_1 \bullet X; A_2 \bullet X; \ldots; A_{m_1} \bullet X)$$

where $A_i \in \mathcal{S}^{n_1}$ for $i = 1, 2, \ldots, m_1$. The linear operators $W : \mathcal{S}^{n_2} \longrightarrow \mathbb{R}^{m_2}$ and $T : \mathcal{S}^{n_2} \longrightarrow \mathbb{R}^{m_1}$ are defined in a similar manner. Problem (2.2.7, 2.2.8) can also be written in a vectorized matrix form (see [25, Section 1]). See Table 2.3 for a summary of the Euclidean Jordan algebraic structure of the positive semidefinite cone in both matrix and vectorized matrix forms. Some applications leading to two-stage SSDPs with recourse can be found in [49].

**Problem 5. Stochastic programming over complex Hermitian positive semidefinite matrices:**

Recall that a square matrix with complex entries is called *Hermitian* if it is equal to its
own conjugate transpose, and that the eigenvalues of a (positive semidefinite) Hermitian matrix are always (nonnegative) real valued.

Let $H^p$ denote the space of complex Hermitian matrices of order $p$. Equipped with the Jordan multiplication $X \circ Y := (XY + YX)/2$, the space $H^p$ forms a Euclidean Jordan algebra. By following a similar argument to that in Example 19 (but using the spectral theorem for complex Hermitian matrices instead of using the spectral theorem for real symmetric matrices, see [43, Theorem 5.4.13]), we can show that the cone of squares of the space of complex Hermitian matrices of order $p$ is the cone of complex Hermitian positive semidefinite matrices of orders $p$, denoted by $H^p_+$. The Euclidean Jordan algebraic structure of the cone of complex Hermitian positive semidefinite matrices is analogous to that of the cone of real symmetric positive semidefinite matrices.

Another subclass of stochastic symmetric programming problems is obtained when $J_1 = H^{n_1}$ and $J_2 = H^{n_2}$, and so we have that $K_{J_1} = H^{n_1}_+$ and $K_{J_2} = H^{n_2}_+$; the cones of complex Hermitian positive semidefinite matrices of orders $n_1$ and $n_2$, respectively. In fact, it is impossible to deal with the field of complex numbers directly as it is an unordered field. However, this can be overcome by defining a transformation that maps complex Hermitian matrices to symmetric matrices; see, for example, [35, 18].

Problem 6. Stochastic programming over quaternion Hermitian positive semidefinite matrices:

We first recall some notions of quaternions and quaternion matrices. The entries of the field of real quaternions has the form $x = x_0 + x_1i + x_2j + x_3k$ where $x_0, x_1, x_2, x_3 \in \mathbb{R}$ and $i, j$, and $k$ are abstract symbols satisfying $i^2 = j^2 = k^2 = ijk = -1$. The conjugate transpose of $x$ is $x^* = x_0 - x_1i - x_2j - x_3k$. If $A$ is a $p \times p$ quaternion matrix and $A^*$ is its conjugate transpose, then $A$ is called Hermitian if $A = A^*$. Note that, if $A$ is Hermitian, then $w^*Aw$ is a real number for any $p$ dimensional column vector $w$ with quaternion components. The matrix $A$ is called positive semidefinite if it is Hermitian
and \( w^* A w \geq 0 \) for any \( p \) dimensional column vector \( w \) with quaternion components. A (positive semidefinite) quaternion Hermitian matrix has (nonnegative) real eigenvalues.

Let \( QH^p \) denote the space of quaternion Hermitian matrices of order \( p \). Equipped with the Jordan multiplication \( X \circ Y := (XY + YX)/2 \), the space \( QH^p \) forms a Euclidean Jordan algebra. It has been shown [46] that the spectral theorem holds also for quaternion matrices (i.e., any quaternion Hermitian matrix is conjugate to a real diagonal matrix). So, by following argument similar to that in Example 19, we can show that the cone of squares of the space of quaternion Hermitian matrices of order \( p \) is the cone of quaternion Hermitian positive semidefinite matrices of orders \( p \), denoted by \( QH^p_+ \). The Euclidean Jordan algebraic structure of the cone of complex Hermitian positive semidefinite matrices is analogous to that of the cone of real symmetric positive semidefinite matrices and that of the cone of complex Hermitian positive semidefinite matrices.

The last class of problems is obtained when \( J_1 = QH^{n_1} \) and \( J_2 = QH^{n_2} \) with \( X \circ Y := (XY + YX)/2 \). Hence \( K_{J_1} = QH^{n_1}_+ \) and \( K_{J_2} = QH^{n_2}_+ \); the cones of quaternion Hermitian positive semidefinite matrices of orders \( n_1 \) and \( n_2 \), respectively. Similar to Problem (5), there is a difficulty in dealing with quaternion entries, but this difficulty can be overcome by defining a transformation that maps Hermitian matrices with quaternion entries to symmetric matrices; see, for example, [35, 18].

Of course, it is also possible to consider a “mixed problem” such as that one obtained by selecting \( J_1 = R^{n_1} \) and \( J_2 = E^{n_2} \), or by selecting \( J_1 = E^{n_1} \) and \( J_2 = S^{n_2} \), etc.

In many applications we find that the models lead to SSPs in dual standard form. The focus of this dissertation is on the SSP in dual standard form. In the next two chapters, we will focus on interior point algorithms for solving the SSP problem (2.1.3) and (2.1.4).
Chapter 3

A Class of Polynomial Logarithmic Barrier Decomposition Algorithms for Stochastic Symmetric Programming

In this chapter, we turn our attention to the study of logarithmic barrier decomposition-based interior point methods for the general SSP problem. Since our algorithm applies to all symmetric cones, this work extends Zhao’s work [48] which focuses particularly on SLP problem (Problem 1), and Mehrotra and Özevin’s work [25] which focuses particularly on the SSDP problem (Problem 4). More specifically, we use logarithmic barrier interior point methods to present a Bender’s decomposition-based algorithm for solving the SSP problem (2.1.3) and (2.1.4) and then prove its polynomial complexity. Our convergence analysis proceeds by showing that the log barrier associated with the recourse function of SSPs behaves as a strongly self-concordant barrier and forms a self-concordant family on the first stage solutions. Our procedure closely follows that of [25] (which essentially follows the procedure of [48]), but our setting is much more general. The results of this
chapter have been submitted for publication [3].

3.1 The log barrier problem for SSPs

We begin by presenting the extensive formulation of the SSP problem (2.1.3, 2.1.4) and the log barrier [30] problems associated with them.

3.1.1 Formulation and assumptions

We now examine (2.1.3, 2.1.4) when the event space \( \Omega \) is finite. Let \( \{(T^{(k)}, W^{(k)}, h^{(k)}, d^{(k)}) : k = 1, 2, \ldots, K\} \) be the set of the possible values of the random variables \( (T(\omega), W(\omega), h(\omega), d(\omega)) \) and let \( p_k = P(T(\omega), W(\omega), h(\omega), d(\omega)) = (T^{(k)}, W^{(k)}, h^{(k)}, d^{(k)}) \) be the associated probability for \( k = 1, 2, \ldots, K \). Then Problem (2.1.3, 2.1.4) becomes

\[
\max \ c^T x + \sum_{k=1}^{K} p_k Q^{(k)}(x) \\
\text{s.t. } A x \preceq_{K,J_1} b,
\]

(3.1.1)

where, for \( k = 1, 2, \ldots, K \), \( Q^{(k)}(x) \) is the maximum value of the problem

\[
\max \ d^{(k)^T} y^{(k)} \\
\text{s.t. } W^{(k)} y^{(k)} \preceq_{K,J_2} h^{(k)} - T^{(k)} x,
\]

(3.1.2)

where \( x \in \mathbb{R}^{m_1} \) is the first-stage decision variable, and \( y^{(k)} \in \mathbb{R}^{m_2} \) is the second-stage variable for \( k = 1, 2, \ldots, K \). Now, for convenience we redefine \( d^{(k)} \) as \( d^{(k)} := p_k d^{(k)} \) for
\( k = 1, 2, \ldots, K \), and rewrite Problem (3.1.1, 3.1.2) as

\[
\max \; c^T x + \sum_{k=1}^{K} Q^{(k)}(x) \\
\text{s.t.} \; A x + s = b \\
\quad \quad s \succeq_{J_1} 0,
\]

(3.1.3)

where, for \( k = 1, 2, \ldots, K \), \( Q^{(k)}(x) \) is the maximum value of the problem

\[
\max \; d^{(k)^T} y^{(k)} \\
\text{s.t.} \; W^{(k)} y^{(k)} + s^{(k)} = h^{(k)} - T^{(k)} x \\
\quad \quad s^{(k)} \succeq_{J_2} 0.
\]

(3.1.4)

Let \( \nu^{(k)} \) be the second-stage dual multiplier. The dual of Problem of (3.1.4) is

\[
\min \; (h^{(k)} - T^{(k)} x) \bullet \nu^{(k)} \\
\text{s.t.} \; W^{(k)^T} \nu^{(k)} = d^{(k)} \\
\quad \quad \nu^{(k)} \succeq_{J_2} 0.
\]

(3.1.5)

The log barrier problem associated with Problem (3.1.3, 3.1.4) is

\[
\max \; \eta(\mu, x) := c^T x + \sum_{k=1}^{K} \rho^{(k)}(\mu, x) + \mu \ln \det s \\
\text{s.t.} \; A x + s = b \\
\quad \quad s \succ_{J_1} 0,
\]

(3.1.6)

where, for \( k = 1, 2, \ldots, K \), \( \rho^{(k)}(\mu, x) \) is the maximum value of the problem

\[
\max \; d^{(k)^T} y^{(k)} + \mu \ln \det s^{(k)} \\
\text{s.t.} \; W^{(k)} y^{(k)} + s^{(k)} = h^{(k)} - T^{(k)} x \\
\quad \quad s^{(k)} \succ_{J_2} 0.
\]

(3.1.7)
(Here \( \mu > 0 \) is a barrier parameter.) If for some \( k \), Problem (3.1.7) is infeasible, then we define \( \sum_{k=1}^{K} \rho(k)(\mu, x) := \infty \). The log barrier problem associated with Problem (3.1.5) is the problem

\[
\begin{align*}
\min & \quad (h^{(k)} - T^{(k)}x) \cdot \nu^{(k)} - \mu \ln \det \nu^{(k)} \\
\text{s.t.} & \quad W^{(k)} \nu^{(k)} = d^{(k)} \\
& \quad \nu^{(k)} \succeq_{K_{J_2}} 0,
\end{align*}
\]

which is the Lagrangian dual of (3.1.7). Because Problems (3.1.7) and (3.1.8) are, respectively, concave or convex, \((y^{(k)}, s^{(k)})\) and \(\nu^{(k)}\) are optimal solutions to (3.1.7) and (3.1.8), respectively, if and only if they satisfy the following optimality conditions:

\[
\begin{align*}
s^{(k)} \circ \nu^{(k)} &= \mu e_2, \\
W^{(k)} y^{(k)} + s^{(k)} &= h^{(k)} - T^{(k)}x, \\
W^{(k)} \nu^{(k)} &= d^{(k)}, \\
s^{(k)} \succeq_{K_{J_2}} 0, \nu^{(k)} \succeq_{K_{J_2}} 0.
\end{align*}
\]

The elements \( s^{(k)} \) and \( \nu^{(k)} \) may not operator commute. So, the equality \( s^{(k)} = \mu \nu^{(k)^{-1}} \) may not hold (see Remark 1 in Chapter 1). In fact, we need to scale the optimality conditions (3.1.9) so that the scaled elements are simultaneously decomposed.

Let \( p \succeq_{K_{J_2}} 0 \). From now on, with respect to \( p \), we define

\[
\tilde{s}^{(k)} := [p^{-1}] s^{(k)}, \quad \tilde{\nu}^{(k)} := [p] \nu^{(k)}, \quad \tilde{h} := [p^{-1}] h, \quad \tilde{W}^{(k)} := [p^{-1}] W^{(k)}, \quad \text{and} \quad \tilde{T}^{(k)} := [p^{-1}] T^{(k)}.
\]

Recall that, \([p][p^{-1}] = [p][p]^{-1} = I\). We have the following lemma and proposition.

**Lemma 3.1.1** (Lemma 28, [36]). Let \( p \) be an invertible element in \( J_2 \). Then \( s \circ \nu = \mu e_2 \) if and only if \( \tilde{s} \circ \tilde{\nu} = \mu e_2 \).

**Proposition 3.1.1.** \((y, s, \nu)\) satisfies the optimality conditions (3.1.1) if and only if
$(y, \tilde{s}, \tilde{\nu})$ satisfies the relaxed optimality conditions:

\[
\begin{align*}
\tilde{s}^{(k)} \circ \tilde{\nu}^{(k)} &= \mu e_2, \\
\tilde{W}^{(k)}y^{(k)} + \tilde{s}^{(k)} &= h^{(k)} - \tilde{T}^{(k)}x, \\
\tilde{W}^{(k)\top} \tilde{\nu}^{(k)} &= d^{(k)}, \\
\tilde{s}^{(k)} &\succ_{K_{J_2}} 0, \quad \tilde{\nu}^{(k)} \succ_{K_{J_2}} 0.
\end{align*}
\]

(3.1.10)

**Proof.** The proof follows from Lemma 3.1.1, Proposition 1.3.2, and the fact that $\lceil p \rceil (K_{J_2}) = K_{J_2}$, and likewise, as an operator, $\lceil p \rceil (\text{int}(K_{J_2})) = \text{int}(K_{J_2})$, because $K_{J_2}$ is symmetric. □

To our knowledge, this is the first time this effective way of scaling is being used for stochastic programing, but it was originally proposed by Monteiro [28] and Zhang [47] for DSDP, and after that generalized by Schmieta and Alizadeh [36] for DSP.

With this change of variable Problem (3.1.6, 3.1.7) becomes

\[
\begin{align*}
\max \quad \tilde{\eta}(\mu, x) := c^\top x + \sum_{k=1}^{K} \tilde{\rho}^{(k)}(\mu, x) + \mu \ln \det s \\
\text{s.t.} \quad & Ax + s = b \\
& s \succ_{K_{J_1}} 0,
\end{align*}
\]

(3.1.11)

where, for $k = 1, 2, \ldots, K$, $\tilde{\rho}^{(k)}(\mu, x)$ is the maximum value of the problem

\[
\begin{align*}
\max \quad d^{(k)\top} y^{(k)} + \mu \ln \det \tilde{s}^{(k)} \\
\text{s.t.} \quad & \tilde{W}^{(k)}y^{(k)} + \tilde{s}^{(k)} = h^{(k)} - \tilde{T}^{(k)}x \\
& \tilde{s}^{(k)} \succ_{K_{J_2}} 0,
\end{align*}
\]

(3.1.12)
and Problem (3.1.8) becomes

\[
\min \ (\tilde{\mathbf{h}}^{(k)} - \tilde{T}^{(k)} \mathbf{x}) \bullet \tilde{\nu}^{(k)} - \mu \ln \det \tilde{\nu}^{(k)} \\
\text{s.t.} \quad \tilde{W}^{(k)^T} \tilde{\nu}^{(k)} = \mathbf{d}^{(k)}
\]

(3.1.13)

Note that the Problem (3.1.7) and Problem (3.1.12) have the same maximizer, but their optimal objective values are equal up to a constant. More precisely, the difference between their optimal objective values would be \(\mu \ln \det \mathbf{p}^2\) (see item 1 of Lemma (1.3.2)).

Similarly, Problems (3.1.8) and (3.1.13) have the same minimizer but their optimal objective values differ by the same value \((\mu \ln \det \mathbf{p}^2)\).

The SSP (3.1.11, 3.1.12) can be equivalently written as a DSP:

\[
\max \ c^T \mathbf{x} + \mu \ln \det \mathbf{s} + \sum_{k=1}^{K} \left( \mathbf{d}^{(k)^T} \tilde{\mathbf{y}}^{(k)} + \mu \ln \det \tilde{\mathbf{s}}^{(k)} \right) \]

(3.1.14)

s.t. \(A \mathbf{x} + \mathbf{s} = \mathbf{b}\)

\[
\tilde{W}^{(k)} \mathbf{y}^{(k)} + \tilde{\mathbf{s}}^{(k)} = \tilde{\mathbf{h}}^{(k)} - \tilde{T}^{(k)} \mathbf{x}, \ k = 1, 2, \ldots, K
\]

\[
\mathbf{s} \succ_{\mathcal{K}_{J_1}} \mathbf{0}, \ \tilde{\mathbf{s}}^{(k)} \succ_{\mathcal{K}_{J_2}} \mathbf{0}, \ k = 1, 2, \ldots, K.
\]

In Subsection 3.1.2, we need to compute \(\nabla_x \tilde{\eta}(\mu, \mathbf{x})\) and \(\nabla^2 \tilde{\eta}(\mu, \mathbf{x})\) so that we can determine the Newton direction defined by

\[
\Delta \mathbf{x} := -\left(\nabla^2 \tilde{\eta}(\mu, \mathbf{x})\right)^{-1} \nabla_x \tilde{\eta}(\mu, \mathbf{x})
\]

for our algorithms. We shall see that each choice of \(\mathbf{p}\) leads to a different search direction (see Algorithm 1). As we mentioned earlier, we are interested in the class of \(\mathbf{p}\) for which the scaled elements are simultaneously decomposed. In view of Theorem 1.3.5, it is enough to choose \(\mathbf{p}\) so that \(\tilde{s}^{(k)}\) and \(\tilde{\nu}^{(k)}\) operator commute. That is, we restrict our attention to
the following set of scalings:

\[ C(s^{(k)},\nu^{(k)}) := \{ p \succ_{K_j} 0 \mid \tilde{s}^{(k)} \text{ and } \tilde{\nu}^{(k)} \text{ operator commute} \}. \]

We introduce the following definition [1].

**Definition 3.1.1.** The set of directions \( \Delta x \) arising from those \( p \in C(s^{(k)},\nu^{(k)}) \) is called the commutative class of directions, and a direction in this class is called a commutative direction.

It is clear that \( p = e \) may not be in \( C(s^{(k)},\nu^{(k)}) \). The following choices of \( p \) shows that the set \( C(s^{(k)},\nu^{(k)}) \) is not empty.

We may choose \( p = s^{(k)1/2} \) and get \( \tilde{s}^{(k)} = e_2 \). In fact, by using Lemma 1.3.2, it can be seen that

\[
\tilde{s}^{(k)} = [p^{-1}] s^{(k)} = [s^{(k)1/2}] (s^{(k)1/2})^2 = \left( \begin{bmatrix} s^{(k)1/2} \\ s^{(k)1/2} \end{bmatrix} \right) e_2 = \left( \begin{bmatrix} s^{(k)1/2} \\ s^{(k)1/2} \end{bmatrix} \right) e_2 = e_2.
\]

We may choose \( p = \nu^{(k)1/2} \) and get \( \tilde{\nu}^{(k)} = e_2 \). To see this, note that by using Lemma 1.3.2 we have

\[
\tilde{\nu}^{(k)} = [p] \nu^{(k)} = [\nu^{(k)1/2}] (\nu^{(k)1/2})^2 = \left( \begin{bmatrix} \nu^{(k)1/2} \\ \nu^{(k)1/2} \end{bmatrix} \right) e_2 = \left( \begin{bmatrix} \nu^{(k)1/2} \\ \nu^{(k)1/2} \end{bmatrix} \right) e_2 = e_2.
\]

The above two choices of directions are well-known in commutative classes. These two choices form a class of Newton directions derived by Helmberg et al [20], Monteiro [28],
and Kojima et al [22], and referred to as the HRVW/KSH/M directions. It is interesting to mention that the popular search direction due to Nesterov-Todd (NT) is also in a commutative class, because in this case one chooses $p$ in such a way that $\tilde{\nu}^{(k)} = \tilde{s}^{(k)}$. More precisely, in the NT direction we choose
\[
p = \left( \left[ \nu^{(k)\frac{1}{2}} \right] \left( \left[ \nu^{(k)\frac{1}{2}} \right] s^{(k)} \right)^{-\frac{1}{2}} \right)^{-\frac{1}{2}} = \left( \left[ s^{(k)\frac{1}{2}} \right] \left( \left[ s^{(k)\frac{1}{2}} \right] \nu^{(k)} \right)^{\frac{1}{2}} \right)^{-\frac{1}{2}},
\]
and, using Lemma 1.3.2, we get
\[
[p]^2 \nu^{(k)} = [p^2] \nu^{(k)} = \left[ \nu^{(k)\frac{1}{2}} \left( \left[ \nu^{(k)\frac{1}{2}} \right] s^{(k)} \right)^{-\frac{1}{2}} \right]^{-1} \nu^{(k)} = \left[ \nu^{(k)\frac{1}{2}} \left( \left[ \nu^{(k)\frac{1}{2}} \right] s^{(k)} \right)^{\frac{1}{2}} \right] \nu^{(k)} = \left[ \left[ \nu^{(k)\frac{1}{2}} \right] \left( \left[ \nu^{(k)\frac{1}{2}} \right] s^{(k)} \right)^{\frac{1}{2}} \right] \left[ \nu^{(k)\frac{1}{2}} \right] \nu^{(k)} = \left[ \left[ \nu^{(k)\frac{1}{2}} \right] \left( \nu^{(k)\frac{1}{2}} \right) \right] e_2 = \left[ \nu^{(k)\frac{1}{2}} \right] s^{(k)} = s^{(k)}.
\]
Thus, $\tilde{\nu}^{(k)} = [p] \nu^{(k)} = [p^{-1}] s^{(k)} = \tilde{s}^{(k)}$.

We proceed by making some assumptions. First we define

\[\mathcal{F}_1 := \{ x : Ax + s = b, \ s \succ_{K,\mathcal{J}_1} 0 \};\]
\[\mathcal{F}^{(k)}(x) := \{ y^{(k)} : \tilde{W}^{(k)} y^{(k)} + \tilde{s}^{(k)} = \tilde{h}^{(k)} - \tilde{T}^{(k)} x, \tilde{s}^{(k)} \succ_{K,\mathcal{J}_2} 0 \} \text{ for } k = 1, 2, \ldots, K;\]
\[\mathcal{F}_2^{(k)} := \{ x : \mathcal{F}^{(k)}(x) \neq \emptyset \} \text{ for } k = 1, 2, \ldots, K;\]
\[\mathcal{F}_2 := \bigcap_{k=1}^K \mathcal{F}_2^{(k)};\]
\[\mathcal{F}_0 := \mathcal{F}_1 \cap \mathcal{F}_2;\]
\[ \mathcal{F} := \{ (x, s, \gamma) \times (y^{(1)}, \tilde{s}^{(1)}, \tilde{\nu}^{(1)}, \ldots, y^{(K)}, \tilde{s}^{(K)}, \tilde{\nu}^{(K)}) : Ax + s = b, s \succ K_{J_1} 0, \]
\[ \tilde{W}^{(k)} y^{(k)} + \tilde{s}^{(k)} = \tilde{h}^{(k)} - \tilde{T}^{(k)} x, \tilde{s}^{(k)} \succ K_{J_2} 0, \tilde{W}^{(k)T} \tilde{\nu}^{(k)} = d^{(k)}, \]
\[ \tilde{\nu}^{(k)} \succ K_{J_2} 0, k = 1, 2, \ldots, K; A^T \gamma + \sum_{k=1}^{K} \tilde{T}^{(k)T} \tilde{\nu}^{(k)} = c \}. \]

Here \( \gamma \) is the first-stage dual multiplier. Now we make

**Assumption 3.1.1.** The matrices \( A \) and \( \tilde{W}^{(k)} \) for all \( k \) have full column rank.

**Assumption 3.1.2.** The set \( \mathcal{F} \) is nonempty.

Assumption 3.1.1 is for convenience. Assumption 3.1.2 guarantees strong duality for first- and second-stage SSPs. In other words, it requires that Problem (3.1.14) and its dual have strictly feasible solutions. This implies that problems (3.1.11-3.1.14) have a unique solutions. Note that for a given \( \mu > 0, \sum_{k=1}^{K} \tilde{\rho}^{(k)}(\mu, x) < \infty \) if and only if \( x \in \mathcal{F}_2 \). Hence, the feasible region for (3.1.11) is described implicitly by \( \mathcal{F}_0 \). Throughout the paper we denote the optimal solution of the first-stage problem (3.1.11) by \( x(\mu) \), and the solutions of the optimality conditions (3.1.10) by \( (y^{(k)}(\mu, x), \tilde{s}^{(k)}(\mu, x), \tilde{\nu}^{(k)}(\mu, x)) \). The following proposition establishes the relationship between the optimal solutions of Problems (3.1.11, 3.1.12) and those of Problem (3.1.14).

**Proposition 3.1.2.** Let \( \mu > 0 \) be fixed. Then \((x(\mu), s(\mu); y^{(1)}(\mu), \tilde{s}^{(1)}(\mu); \ldots; y^{(K)}(\mu), \tilde{s}^{(K)}(\mu))\) is the optimal solution of (3.1.14) if and only if \((x(\mu), s(\mu))\) is the optimal solution of (3.1.11) and \((y^{(1)}(\mu), \tilde{s}^{(1)}(\mu); \ldots; y^{(K)}(\mu), \tilde{s}^{(K)}(\mu))\) are the optimal solutions for (3.1.12) for given \( \mu \) and \( x = x(\mu) \).

### 3.1.2 Computation of \( \nabla_x \tilde{\eta}(\mu, x) \) and \( \nabla^2_{xx} \tilde{\eta}(\mu, x) \)

In order to compute \( \nabla_x \tilde{\eta}(\mu, x) \) and \( \nabla^2_{xx} \tilde{\eta}(\mu, x) \), we need to determine the derivative of \( \tilde{\rho}^{(k)}(\mu, x) \) with respect to \( x \). Let \((y^{(k)}, \tilde{\nu}^{(k)}, \tilde{s}^{(k)}) := (y^{(k)}(\mu, x), \tilde{\nu}^{(k)}(\mu, x), \tilde{s}^{(k)}(\mu, x)) \). We
First note that from (3.1.10) we have that
\[
(\tilde{h}^{(k)} - \tilde{T}^{(k)} x) \bullet \tilde{\nu}^{(k)} = (\tilde{W}^{(k)} y^{(k)} + \tilde{s}^{(k)}) \bullet \tilde{\nu}^{(k)} = y^{(k)} T (\tilde{W}^{(k)} T \tilde{\nu}^{(k)}) + \tilde{s}^{(k)} \bullet \tilde{\nu}^{(k)} = y^{(k)} T d^{(k)} + r_2 \mu,
\]
where in the second equality we used the observation that
\[
\tilde{W}^{(k)} y^{(k)} \bullet \tilde{\nu}^{(k)} = \sum_{i=1}^{m_2} (y_i^{(k)} \tilde{w}_i^{(k)}) \bullet \tilde{\nu}^{(k)} = \sum_{i=1}^{m_2} y_i^{(k)} (\tilde{w}_i^{(k)} \bullet \tilde{\nu}^{(k)}) = y^{(k)} T (\tilde{W}^{(k)} T \tilde{\nu}^{(k)})
\]
(here \(\tilde{w}_i^{(k)} \in J_2\) is the \(i^{th}\) column of \(\tilde{W}^{(k)}\), and in the last equality we used that \(\text{trace}(e_2) = \text{rank}(J_2) = r_2\). This implies that
\[
(\tilde{h}^{(k)} - \tilde{T}^{(k)} x) \bullet \tilde{\nu}^{(k)} - \mu \ln \det \tilde{\nu}^{(k)} = \tilde{\rho}^{(k)}(\mu, x) - \mu \ln \det \tilde{s}^{(k)} + r_2 \mu - \mu \ln \det \tilde{\nu}^{(k)}
= \tilde{\rho}^{(k)}(\mu, x) + r_2 \mu - \mu \ln \det (\tilde{s}^{(k)} \circ \tilde{\nu}^{(k)})
= \tilde{\rho}^{(k)}(\mu, x) + r_2 \mu (1 - \ln \mu).
\]
Thus,
\[
\tilde{\rho}^{(k)}(\mu, x) = (\tilde{h}^{(k)} - \tilde{T}^{(k)} x) \bullet \tilde{\nu}^{(k)} - \mu \ln \det \tilde{\nu}^{(k)} - r_2 \mu (1 - \ln \mu).
\]  
(3.1.15)

Differentiating (3.1.15) and using the optimality conditions (3.1.10), we obtain
\[
\nabla x \tilde{\rho}^{(k)}(\mu, x) = \nabla x ((\tilde{h}^{(k)} - \tilde{T}^{(k)} x) \bullet \tilde{\nu}^{(k)}) - \mu \nabla x \ln \det \tilde{\nu}^{(k)}
= (\nabla x (\tilde{h}^{(k)} - \tilde{T}^{(k)} x)) T \tilde{\nu}^{(k)} + (\nabla x \tilde{\nu}^{(k)}) T (\tilde{h}^{(k)} - \tilde{T}^{(k)} x) - \mu (\nabla x \tilde{\nu}^{(k)}) T \tilde{\nu}^{(k)}
= -\tilde{T}^{(k)} T \tilde{\nu}^{(k)} + (\nabla x \tilde{\nu}^{(k)}) T (\tilde{W}^{(k)} y^{(k)} + \tilde{s}^{(k)}) - \mu (\nabla x \tilde{\nu}^{(k)}) T \tilde{\nu}^{(k)}
= -\tilde{T}^{(k)} T \tilde{\nu}^{(k)} + (\nabla x \tilde{\nu}^{(k)}) T (\tilde{W}^{(k)} y^{(k)}) + (\nabla x \tilde{\nu}^{(k)}) T (\tilde{s}^{(k)} - \mu \tilde{\nu}^{(k)}
= -\tilde{T}^{(k)} T \tilde{\nu}^{(k)} + (\nabla x \tilde{\nu}^{(k)}) T (\tilde{W}^{(k)} y^{(k)})
\]

From (3.1.10) we have that \(\tilde{W}_i^{(k)} \bullet \tilde{\nu}^{(k)} = d_i^{(k)}\), where \(d_i^{(k)}\) is the \(i^{th}\) entry of \(d^{(k)}\). This implies that

51
\[
(\nabla_x \tilde{\nu}^{(k)})^T (\tilde{W}^{(k)} y^{(k)}) = (\nabla_x \tilde{\nu}^{(k)})^T \sum_{i=1}^{m_2} y_i^{(k)} \tilde{w}_i^{(k)}
\]
\[
= \sum_{i=1}^{m_2} y_i^{(k)} (\nabla_x \tilde{\nu}^{(k)})^T \tilde{w}_i^{(k)}
\]
\[
= \sum_{i=1}^{m_2} y_i^{(k)} \nabla_x (\tilde{\nu}^{(k)} \bullet \tilde{w}_i^{(k)})
\]
\[
= 0.
\]

Thus,
\[
\nabla_x \tilde{\rho}^{(k)}(\mu, x) = -\tilde{T}^{(k)T} \tilde{\nu}^{(k)} \quad \text{and} \quad \nabla^2_{xx} \tilde{\rho}^{(k)}(\mu, x) = -\tilde{T}^{(k)T} \nabla_x \tilde{\nu}^{(k)}.
\]

Therefore, we also need to determine the derivative of \( \tilde{\nu}^{(k)} \) with respect to \( x \). Differentiating (3.1.10) with respect to \( x \), we get the system
\[
\nabla_x \tilde{\nu}^{(k)} = -\mu \left[ s^{(k) -1/2} \right] \nabla_x \tilde{s}^{(k)},
\]
\[
\tilde{W}^{(k)} \nabla_x y^{(k)} + \nabla_x \tilde{s}^{(k)} = -\tilde{T}^{(k)},
\]
\[
\tilde{W}^{(k)T} \nabla_x \tilde{\nu}^{(k)} = 0.
\]

Solving the system (3.1.16), we obtain
\[
\nabla_x \tilde{s}^{(k)} = -\left[ s^{(k)1/2} \right] P^{(k)} \left[ s^{(k) -1/2} \right] \tilde{T}^{(k)},
\]
\[
\nabla_x \tilde{\nu}^{(k)} = \mu \left[ s^{(k) -1/2} \right] P^{(k)} \left[ s^{(k) -1/2} \right] \tilde{T}^{(k)},
\]
\[
\nabla_x y^{(k)} = -R^{(k) -1} \tilde{W}^{(k)T} \left[ s^{(k) -1} \right] \tilde{T}^{(k)},
\]

where
\[
R^{(k)} := R^{(k)}(\mu, x) = \tilde{W}^{(k)T} \left[ s^{(k) -1} \right] \tilde{W}^{(k)},
\]
\[
P^{(k)} := P^{(k)}(\mu, x) = I - \left[ s^{(k) -1/2} \right] \tilde{W}^{(k)} R^{(k) -1} \tilde{W}^{(k)T} \left[ s^{(k) -1/2} \right].
\]

Observing that, by differentiating \( A x + s = b \) with respect to \( x \), we get \( \nabla_x s = -A \).
We then have
\[
\nabla_x \tilde{\eta}(\mu, x) = c - \sum_{k=1}^{K} \nabla_x \tilde{\rho}^{(k)}(\mu, x) + \mu (\nabla_x s)^T s^{-1},
\]
(3.1.19)
and
\[
\nabla^2_{xx} \tilde{\eta}(\mu, x) = -\sum_{k=1}^{K} \tilde{T}^{(k)} \nabla_x \tilde{\nu}^{(k)} + \mu A^T [s^{-1}] \nabla_x s
\]
\[
= -\mu \sum_{k=1}^{K} \tilde{T}^{(k)} \left[ \tilde{s}^{(k)-1/2} \right] P^{(k)} \left[ \tilde{s}^{(k)-1/2} \right] \tilde{T}^{(k)} - \mu A^T [s^{-1}] A.
\]
(3.1.20)

3.2 Self-concordance properties of the log-barrier recourse

The notion of so called self-concordant functions introduced by Nesterov and Nemirovskii [30] allows us to develop polynomial time path following interior point methods for solving SSPs. In this section, we prove that the recourse function with log barrier is a strongly self-concordant function and leads to a strongly self-concordant family with appropriate parameters.

3.2.1 Self-concordance of the recourse function

This subsection is devoted to show that \( \tilde{\eta}(\mu, \cdot) \) is a \( \mu \) strongly self-concordant barrier on \( \mathcal{F}_0 \). First, we have the following definition.

**Definition 3.2.1** (Nesterov and Nemirovskii [30, Definition 2.1.1]). Let \( E \) be a finite-dimensional real vector space space, \( G \) be an open nonempty convex subset of \( E \), and let
be a $C^3$, convex mapping from $G$ to $\mathbb{R}$. Then $f$ is called $\alpha$-self-concordant on $G$ with the parameter $\alpha > 0$ if for every $x \in G$ and $h \in E$, the following inequality holds:

$$|\nabla^2_{xxx} f(x)[h, h, h]| \leq 2\alpha^{-1/2}(\nabla^2_{xx} f(x)[h, h])^{3/2}. \quad (3.2.1)$$

An $\alpha$-self-concordant function $f$ on $G$ is called strongly $\alpha$-self-concordant if $f$ tends to infinity for any sequence approaching a boundary point of $G$.

We note that in the above definition the set $G$ is assumed to be open. However, relative openness would be sufficient to apply the definition. See also [30, Item A, Page 57].

The proof of strongly self-concordance of the function $\tilde{\eta} (\mu, \cdot)$ relies on two lemmas that we state and prove below. We point out that the proof of the first lemma is shown in [30, Proposition 5.4.5] for $s \in S_{n_1}^{++} := \text{int}(S_{n_1}^{++})$, i.e., $s$ lies in the interior of the cone of real symmetric positive semidefinite matrices of orders $n_1$. We now generalize this proof for the case where $s$ lies in the interior of an arbitrary symmetric cone of dimension $n_1$, where $n_1$, as mentioned before, is any positive integer.

**Lemma 3.2.1.** For any fixed $\mu > 0$, the function $f(s) := -\mu \ln \det s$ is a $\mu$ strongly self-concordant barrier on $K_{J_1}$.

**Proof.** Let $\{s_i\}_{i=1}^{\infty}$ be any sequence in $K_{J_1}$. It is clear that the function $f(s_i)$ tends to $\infty$ when $s_i$ approaches a point from boundary of $K_{J_1}$. It remains to show that (3.2.1) is satisfied for $f(s)$ on $K_{J_1}$. Let $s \succ_{K_{J_1}} 0$ and $h \in J_1$, then there exists an element $u \in J_1$ such that $s = u^2$, and therefore, by Lemmas 1.3.1 and 1.3.2, we have that

$$\nabla_s f(s)[h] = -\mu \{ \ln \det (s + th) \}'|_{t=0}$$

$$= -\mu \{ \ln \det (u^2 + th) \}'|_{t=0}$$

$$= -\mu \{ \ln \det ([u](e_1 + t[u]^{-1}h)) \}'|_{t=0}$$

$$= -\mu \{ \ln \det (u^2) + \ln \det (e_1 + t[u]^{-1}h) \}'|_{t=0}$$
\[ -\mu \frac{\text{trace}([u]^{-1}h)}{\det e_1} = -\mu (s^{-1} \cdot h), \]

\[ \nabla^2_{ss} f(s)[h, h] = \nabla_s(\nabla_s f(s)[h]) \cdot h \]
\[ = -\mu (\nabla_s(s^{-1} \cdot h)) \cdot h \]
\[ = -\mu (D_s(s^{-1} \circ h)) \cdot h \]
\[ = \mu (s^{-2} \circ h) \cdot h \]
\[ = \mu \{(s^{-1} \circ h) \cdot (s^{-1} \circ h)\}, \]

\[ \nabla^3_{sss} f(s)[h, h, h] = \nabla_s(\nabla^2_{ss} f(s)[h, h]) \cdot h \]
\[ = \mu (\nabla_s((s^{-1} \circ h) \cdot (s^{-1} \circ h))) \cdot h \]
\[ = \mu (D_s((s^{-1} \circ h) \circ (s^{-1} \circ h))) \cdot h \]
\[ = -2\mu ((s^{-2} \circ h) \circ (s^{-1} \circ h)) \cdot h \]
\[ = -2\mu \{(s^{-1} \circ h) \cdot ((s^{-1} \circ h) \circ (s^{-1} \circ h))\}. \]

Let \( \hat{h} = s^{-1} \circ h \in J_1 \), and \( \lambda_1, \lambda_2, \ldots, \lambda_p \) be its eigenvalues. The result is established by observing that

\[ 2\mu \left| \text{trace}(\hat{h}^2) \right| = 2\mu \sum_{i=1}^p |\lambda_i|^3 \leq 2\mu \left\{ \sum_{i=1}^p \lambda_i^2 \right\}^{3/2} = 2\mu \left\{ \text{trace}(\hat{h}^2) \right\}^{3/2}. \]

**Lemma 3.2.2.** For any fixed \( \mu > 0 \), \( \overline{\rho}^{(k)}(\mu, x) \) is a \( \mu \) strongly self-concordant barrier on \( \mathcal{F}_2^{(k)}, k = 1, 2, \ldots, K \).

**Proof.** Let \( \{x_i\}_{i=1}^\infty \) be any sequence in \( \mathcal{F}_2^{(k)} \). It is clear that the function \( \overline{\rho}^{(k)}(\mu, x_i) \) tends to \( \infty \) when \( x_i \) approaches a point from boundary of \( \mathcal{F}_2^{(k)} \). It remains to show that (3.2.1) is satisfied for \( \overline{\rho}^{(k)}(\mu, x) \) on \( \mathcal{F}_2^{(k)} \). For any \( \mu > 0, x \in \mathcal{F}_2^{(k)} \), and \( d \in \mathbb{R}^{m_1} \), we define the univariate function

\[ \Psi^{(k)}(t) := \nabla^2_{xx}\overline{\rho}^{(k)}(\mu, x + td)[d, d]. \]
Note that $\Psi^{(k)}(0) = \nabla_{xx}^2 \tilde{\rho}^{(k)}(\mu, x)[d, d]$ and $\Psi^{(k)}(0)' = \nabla_{xxx}^3 \tilde{\rho}^{(k)}(\mu, x)[d, d, d]$. So, to prove the lemma, it is enough to show that

$$|\Psi^{(k)}(0)'| \leq \frac{2}{\sqrt{\mu}} |\Psi^{(k)}(0)|^{3/2}.$$  

Let $(\tilde{\nu}^{(k)}(t), \tilde{s}^{(k)}(t), P^{(k)}(t), R^{(k)}(t)) := (\tilde{\nu}^{(k)}(\mu, x+t d), \tilde{s}^{(k)}(\mu, x+t d), P^{(k)}(\mu, x+t d), R^{(k)}(\mu, x+t d))$. We also define

$$u^{(k)}(t) := \sqrt{\mu} P^{(k)}(t) \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix} T^{(k)}(t) d.$$  

By the notations introduced in §4, we have $(\tilde{\nu}^{(k)}, \tilde{s}^{(k)}) = (\tilde{\nu}^{(k)}(0), \tilde{s}^{(k)}(0))$. We also let $(P^{(k)}, R^{(k)}, u^{(k)}) := (P^{(k)}(0), R^{(k)}(0), u^{(k)}(0))$.

Notice that, by the definition of $P^{(k)}$, we have $P^{(k)}^2 = P^{(k)}$. Using (3.1.17) and (3.1.18), we get

$$\Psi^{(k)}(0) = \nabla_{xx}^2 \tilde{\rho}^{(k)}(\mu, x)[d, d]$$

$$= -(\tilde{T}^{(k)} T \nabla_x \tilde{\nu}^{(k)})[d, d]$$

$$= -\mu \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix} P^{(k)} \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix} \tilde{T}^{(k)} [d, d]$$

$$= -\mu d^T (\tilde{T}^{(k)} T \tilde{s}^{(k)-1/2}(t) P^{(k)} \tilde{s}^{(k)-1/2}(t) \tilde{T}^{(k)}) d$$

$$= -u^{(k)} \cdot u^{(k)}$$

$$= -\|u^{(k)}\|^2.$$  

Hence $\Psi^{(k)}(0)' = -2u^{(k)} \cdot u^{(k)}$. So, in order to bound $|\Psi^{(k)}(0)'|$, we need to compute the derivative of $u^{(k)}$ with respect to $t$. Using (3.1.18), we have

$$u^{(k)}' = \sqrt{\mu} \left\{ P^{(k)} \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix} \right\}' T^{(k)} d$$

$$= \sqrt{\mu} \left\{ \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix} - \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix} \tilde{W}^{(k)} R^{(k)} \tilde{W}^{(k)} T \begin{bmatrix} \tilde{s}^{(k)-1/2}(t) \end{bmatrix}^2 \right\}' T^{(k)} d$$

56
\[
\begin{align*}
\mathbf{u}^{(k)} \cdot \mathbf{s}^{(k^{-1/2})} & = (P^{(k)} \mathbf{s}^{(k^{-1/2})} T^{(k)}) \mathbf{d} \cdot \mathbf{s}^{(k^{-1/2})} \mathbf{W}^{(k)} \mathbf{x} \\
& = (\mathbf{W}^{(k)} \mathbf{s}^{(k^{-1/2})} T^{(k)}) \mathbf{d} \cdot P^{(k)} \mathbf{s}^{(k^{-1/2})} \mathbf{W}^{(k)} \mathbf{x} \\
& = \mathbf{u}^{(k)}.
\end{align*}
\]

Notice that, for any \( \mathbf{\xi} \in \mathbb{R}^{m_2} \), we have
\[\begin{align*}
\Psi^{(k)}(0)' &= 2u^{(k)} \bullet u^{(k)'} = 2u^{(k)} \bullet \left( \left[ \tilde{s}^{(k)}^{-1/2}(t) \right]' \left[ \tilde{s}^{(k)}^{-1/2}(t) \right]^{-1} u^{(k)} \right).
\end{align*}\] (3.2.2)

By (3.2.2), (3.1.17) and (3.1.18) and using norm inequalities, we get
\[ = 2\mu^{-1/2}\|u^{(k)}\|^2\|u^{(k)}\| \]
\[ = 2\mu^{-1/2}\|\Psi^{(k)}(0)\|^{3/2}. \] The lemma is established. \(\square\)

**Theorem 3.2.1.** For any fixed \(\mu > 0\), \(\tilde{\eta}(\mu, x)\) is a \(\mu\) strongly self-concordant barrier on \(F_0\).

**Proof.** It is trivial to see that the linear function \(c^T x\) is a \(\mu\) strongly self-concordant barrier on \(F_1\) (indeed, both sides of (3.2.1) are identically zero). By Lemma 3.2.1, one can see that the function \(\mu \ln \det s\) is also a \(\mu\) strongly self-concordant barrier on \(F_1\).

By Lemma 3.2.2 and [30, Proposition 2.1.1(ii)], we can conclude that \(\sum_{k=1}^{K} \tilde{\rho}^{(k)}(\mu, x)\) is a \(\mu\) strongly self-concordant barrier on \(F_2\). The theorem is then established by [30, Proposition 2.1.1(ii)]. \(\square\)

### 3.2.2 Parameters of the self-concordant family

We have shown that the recourse function \(\tilde{\eta}(\mu, \cdot)\) is a strongly self-concordant function. Having such a property, this function enjoys many nice features. In this subsection, we show that the family of functions \(\{\tilde{\eta}(\mu, \cdot) : \mu > 0\}\) is a strongly self-concordant family with appropriate parameters. We first introduce the definition of self-concordant families.

**Definition 3.2.2** (Nesterov and Nemirovskii [30, Definition 3.1.1]). Let \(\mathbb{R}_{++}\) be the set of all positive real numbers. Let \(G\) be an open nonempty convex subset of \(\mathbb{R}^n\). Let \(\mu \in \mathbb{R}_{++}\) and let \(f_\mu : \mathbb{R}_{++} \times G \to \mathbb{R}\) be a family of functions indexed by \(\mu\). Let \(\alpha_1(\mu), \alpha_2(\mu), \alpha_3(\mu), \alpha_4(\mu)\) and \(\alpha_5(\mu) : \mathbb{R}_{++} \to \mathbb{R}_{++}\) be continuously differentiable functions on \(\mu\). Then the family of functions \(\{f_\mu\}_{\mu \in \mathbb{R}_{++}}\) is called strongly self-concordant with the parameters \(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5\), if the following conditions hold:

(i) \(f_\mu\) is continuous on \(\mathbb{R}_{++} \times G\), and for fixed \(\mu \in \mathbb{R}_{++}\), \(f_\mu\) is convex on \(G\). \(f_\mu\) has three partial derivatives on \(G\), which are continuous on \(\mathbb{R}_{++} \times G\) and continuously differentiable with respect to \(\mu\) on \(\mathbb{R}_{++}\).
(ii) For any \( \mu \in \mathbb{R}_{++} \), the function \( f_\mu \) is strongly \( \alpha_1(\mu) \)-self-concordant.

(iii) For any \((\mu, x) \in \mathbb{R}_{++} \times G \) and any \( h \in \mathbb{R}^n \),

\[
\{\nabla_x f_\mu(\mu, x)[h]\}' - \{\ln \alpha_3(\mu)\}' \nabla_x f_\mu(\mu, x)[h] \leq \alpha_4(\mu) \alpha_1(\mu)^{\frac{3}{2}} \left( \nabla_{xx}^2 f_\mu(\mu, x)[h, h] \right)^{\frac{1}{2}},
\]

\[
\{\nabla_{xx}^2 f_\mu(\mu, x)[h, h]\}' - \{\ln \alpha_2(\mu)\}' \nabla_{xx}^2 f_\mu(\mu, x)[h, h] \leq 2\alpha_5(\mu) \nabla_{xx}^2 f_\mu(\mu, x)[h, h].
\]

In this section, we need to compute \( \{\nabla_x \tilde{\eta}(\mu, x)\}' \), and in order to do so we need to determine the partial derivative of \( \tilde{\nu}^{(k)}(\mu, x) \) with respect to \( \mu \). Let \((y^{(k)}', \tilde{\nu}^{(k)'}, \tilde{s}^{(k)'}) \) denote the partial derivatives of \((y^{(k)}(\mu, x), \tilde{\nu}^{(k)}(\mu, x), \tilde{s}^{(k)}(\mu, x)) \) with respect to \( \mu \). Differentiating (3.1.10) with respect to \( \mu \), we get the system

\[
\tilde{s}^{(k)} \circ \tilde{\nu}^{(k)'} + \tilde{\nu}^{(k)} \circ \tilde{s}^{(k)'} = e_2,
\]

\[
W^{(k)} y^{(k)'} + \tilde{s}^{(k)'} = 0, \tag{3.2.3}
\]

\[
W^{(k)T} \tilde{\nu}^{(k)'} = 0.
\]

Solving the system (3.2.3), we obtain

\[
\tilde{s}^{(k)'} = W^{(k)} R^{(k)-1} W^{(k)T} \tilde{s}^{(k)'}^{-1},
\]

\[
\tilde{\nu}^{(k)'} = \left[ \tilde{s}^{(k)}^{-\frac{1}{2}}(t) \right] P^{(k)} e_2, \tag{3.2.4}
\]

\[
y^{(k)'} = -R^{(k)-1} W^{(k)T} \tilde{s}^{(k)'}^{-1}.
\]

The proof of strongly self-concordance of the family \( \{\tilde{\eta}(\mu, \cdot) : \mu > 0\} \) depends on the following two lemmas.

**Lemma 3.2.3.** For any \( \mu > 0, x \in \mathcal{F}_0 \) and \( h \in \mathbb{R}^{m_1} \), the following inequality holds:

\[
|\{\nabla_x \tilde{\eta}(\mu, x)^T[h]\}'| \leq \left( \frac{-(r_1 + Kr_2)}{\mu} \nabla_{xx}^2 \tilde{\eta}(\mu, x)[h, h] \right)^{1/2}.
\]
Proof. By differentiating (3.1.19) with respect to \( \mu \) and applying (3.2.4), we obtain

\[
\{ \nabla_x \tilde{\eta}(\mu, x) \}' = -\sum_{k=1}^{K} \tilde{T}^{(k)T} \left[ \begin{array}{c} \tilde{s}^{(k)1/2}(t) \end{array} \right] P^{(k)} e_2 - A^T s^{-1} = -\frac{1}{\sqrt{\mu}} B \varepsilon,
\]

where \( B \in \mathbb{R}^{m_1 \times (K_n_2 + n_1)} \) is defined by

\[
B := \left[ \sqrt{\mu} T^{(1)T} \left[ \begin{array}{c} \tilde{s}^{(1)1/2}(t) \end{array} \right] P^{(1)}, \ldots, \sqrt{\mu} T^{(K)T} \left[ \begin{array}{c} \tilde{s}^{(K)1/2}(t) \end{array} \right] P^{(K)} \right], \sqrt{\mu} A^T \left[ s^{-1/2} \right]
\]

and

\[
\varepsilon := (e_2; \ldots; e_2; e_1) \in J_2 \times \cdots \times J_2 \times J_1.
\]

Notice that, in view of (3.1.20), we have

\[
BB^T = \mu \sum_{k=1}^{K} \tilde{T}^{(k)T} \left[ \begin{array}{c} \tilde{s}^{(k)1/2}(t) \end{array} \right] P^{(k)} \left[ \begin{array}{c} \tilde{s}^{(k)1/2}(t) \end{array} \right] T^{(k)} + \mu A^T \left[ s^{-1} \right] A = -\nabla_{xx}^2 \tilde{\eta}(\mu, x).
\]

This gives

\[
-\{ \nabla_x \tilde{\eta}(\mu, x)^T \}' \{ \nabla_{xx}^2 \tilde{\eta}(\mu, x) \}^{-1} \{ \nabla_x \tilde{\eta}(\mu, x) \}' = \frac{1}{\mu} \varepsilon \bullet B^T [BB^T]^{-1} B \varepsilon
\]

\[
\leq \frac{1}{\mu} \varepsilon \bullet \varepsilon
\]

\[
= \frac{1}{\mu} (e_1 \bullet e_1 + K e_2 \bullet e_2).
\]

Recall that \( e_1 \bullet e_1 = \text{rank}(J_1) = r_1 \) and \( e_2 \bullet e_2 = \text{rank}(J_2) = r_2 \). It follows that

\[
-\{ \nabla_x \tilde{\eta}(\mu, x)^T \}' \{ \nabla_{xx}^2 \tilde{\eta}(\mu, x) \}^{-1} \{ \nabla_x \tilde{\eta}(\mu, x) \}' \leq \frac{(r_1 + Kr_2)}{\mu}. \tag{3.2.5}
\]

We then have

\[
\left| \{ \nabla_x \tilde{\eta}(\mu, x)^T [h] \}' \right| \leq \left( -\{ \nabla_x \tilde{\eta}(\mu, x)^T \}' \{ \nabla_{xx}^2 \tilde{\eta}(\mu, x) \}^{-1} \{ \nabla_x \tilde{\eta}(\mu, x) \}' \right)^{1/2} \left( -\nabla_{xx}^2 \tilde{\eta}(\mu, x) [h, h] \right)^{1/2}
\]

\[
\leq \left( \frac{(r_1 + Kr_2)}{\mu} \nabla_{xx}^2 \tilde{\eta}(\mu, x) [h, h] \right)^{1/2}, \text{ as desired. } \square
\]

61
Lemma 3.2.4. For any $\mu > 0$, $x \in F_0$ and $h \in \mathbb{R}^{m_1}$, the following inequality holds:

$$\left| \{ \nabla^2 \overline{\bar{\eta}}(\mu, x)[h, h] \} \right| \leq \frac{2\sqrt{2}}{\mu} \nabla^2 \overline{\bar{\eta}}(\mu, x)[h, h].$$

Proof. Let $(\tilde{\nu}^{(k)}, \tilde{s}^{(k)}, P^{(k)}, R^{(k)}) := (\tilde{\nu}^{(k)}(\mu, x), \tilde{s}^{(k)}(\mu, x), P^{(k)}(\mu, x), R^{(k)}(\mu, x))$. We fix $h \in \mathbb{R}^{m_1}$ and define

$$u^{(k)} := \sqrt{\mu} P^{(k)} \left[ \tilde{s}^{(k)\rightarrow 1/2} \right] T^{(k)} h.$$ 

Now by following some similar steps as in the proof of Lemma 3.2.2, and using (3.2.2), (3.1.18), (3.2.3), (3.1.10), and (3.2.4), we get

$$(u^{(k)} \bullet u^{(k)})' = 2 u^{(k)} \bullet u^{(k)}'$$

$$= 2 u^{(k)} \bullet \left( \tilde{s}^{(k)\rightarrow 1/2}(t) \left[ \tilde{s}^{(k)\rightarrow 1/2}(t) \right]^{-1} u^{(k)} \right)$$

$$= u^{(k)} \bullet \left[ \tilde{s}^{(k)\rightarrow 1/2}(t) \left[ \tilde{s}^{(k)\rightarrow 1/2}(t) \right]' \left[ \tilde{s}^{(k)\rightarrow 1/2}(t) \right] u^{(k)} \right]$$

$$= u^{(k)} \bullet \left[ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \left[ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \right]' \left[ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \right] u^{(k)} \right]$$

$$= 2 u^{(k)} \bullet \left[ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \left[ \tilde{\nu}^{(k)}(t), \tilde{\nu}^{(k)}(t)' \right] \tilde{\nu}^{(k)\rightarrow 1/2}(t) u^{(k)} \right]$$

$$= 2 u^{(k)} \bullet \left[ e_2, \tilde{\nu}^{(k)\rightarrow 1/2}(t) \circ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \right] u^{(k)}$$

$$= 2 u^{(k)} \bullet \left[ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \circ \tilde{\nu}^{(k)\rightarrow 1/2}(t) u^{(k)} \right]$$

$$= 2 \mu^{-1} u^{(k)} \bullet \left[ \tilde{s}^{(k)}(t) \circ \tilde{\nu}^{(k)\rightarrow 1/2}(t) \right] u^{(k)}$$

$$= 2 \mu^{-1} u^{(k)} \bullet \left[ e_2 - \tilde{\nu}^{(k)}(t) \circ \tilde{s}^{(k)\rightarrow 1/2}(t) \right] u^{(k)}$$

$$= 2 \mu^{-1} u^{(k)} \bullet \left[ e_2 - \mu \tilde{s}^{(k)\rightarrow 1/2}(t) \circ \tilde{s}^{(k)\rightarrow 1/2}(t) \right] u^{(k)}$$

$$\leq 2 \mu^{-1} \| u^{(k)} \|^2 \left\| e_2 - \mu \left[ \tilde{s}^{(k)\rightarrow 1/2}(t) \right] \tilde{s}^{(k)\rightarrow 1/2}(t) \right\|$$

$$= 2 \mu^{-1} \| u^{(k)} \|^2 \left\| e_2 - \mu \left[ \tilde{s}^{(k)\rightarrow 1/2}(t) \right] \tilde{W}^{(k)} R^{(k)\rightarrow 1/2} \tilde{W}^{(k)\rightarrow 1/2} \right\|$$

$$= 2 \mu^{-1} \| u^{(k)} \|^2 \| e_2 - \mu (e_2 - P^{(k)} e_2) \|$$

$$\leq 2 \mu^{-1} \| u^{(k)} \|^2 \| e_2 \|,$$

where the last inequality is obtained by observing that $e_2 - P^{(k)} e_2 \succeq \kappa_{J_2} 0$, which can be
immediately seen by noting that $P^{(k)} \preceq I$.

Recall that $\|e_2\|^2 = \text{rank}(J_2) = r_2$. This gives us

$$|(u^{(k)} \cdot u^{(k)})'| \leq \frac{2\sqrt{r_2}}{\mu} u^{(k)} \cdot u^{(k)}.$$

From (3.1.20), we have that

$$\nabla^2_{xx} \tilde{\eta}(\mu, x)[h, h] = -\sum_{k=1}^{K} u^{(k)} \cdot u^{(k)} - \mu \left( A^T \left[ s^{-1} \right] A \right)[h, h].$$

Therefore, for any $h \in \mathbb{R}^n$, we have

$$|\{\nabla^2_{xx} \tilde{\eta}(\mu, x)[h, h]\}'| \leq \sum_{k=1}^{K} |(u^{(k)} \cdot u^{(k)})'| + \mu \left( A^T \left[ s^{-1} \right] A \right)[h, h]$$

$$\leq \frac{2\sqrt{r_2}}{\mu} \sum_{k=1}^{K} u^{(k)} \cdot u^{(k)} + \left( A^T \left[ s^{-1} \right] A \right)[h, h]$$

$$\leq -\frac{2\sqrt{r_2}}{\mu} \nabla^2_{xx} \tilde{\eta}(\mu, x)[h, h].$$

This completes the proof. □

**Theorem 3.2.2.** The family $\{\tilde{\eta}(\mu, \cdot) : \mu > 0\}$ is a strongly self-concordant family with the following parameters

$$\alpha_1(\mu) = \mu, \; \alpha_2(\mu) = \alpha_3(\mu) = 1, \; \alpha_4(\mu) = \frac{\sqrt{r_1 + Kr_2}}{\mu}, \; \alpha_5(\mu) = \frac{\sqrt{r_2}}{\mu}.$$

**Proof.** It is clear that condition (i) of Definition 3.2.2 holds. Theorem 3.2.1 shows that condition (ii) is satisfied and Lemmas 3.2.4 and 3.2.3 show that condition (iii) is satisfied. □
3.3 A class of logarithmic barrier algorithms for solving SSPs

In §3.2 we have established that the parametric functions \( \widetilde{\eta}(\mu, \cdot) \) constitute a strongly self-concordant family. Therefore, it is straightforward to develop primal path following interior point algorithms for solving SSP (3.1.3, 3.1.4). In this section we introduce a class of log barrier algorithms for solving this problem. This class is stated formally in Algorithm 1.

**Algorithm 1 The Decomposition Algorithm for Solving SSP**

**Require:** \( \epsilon > 0, \gamma \in (0, 1), \theta > 0, \beta > 0, x^0 \in \mathcal{F}_0 \) and \( \mu^0 > 0 \).

\[
x := x^0, \quad \mu := \mu^0
\]

while \( \mu \geq \epsilon \) do

\[
\text{for } k = 1, 2, \ldots, K \text{ do}
\]

solve (3.1.9) to obtain \( (y^{(k)}, s^{(k)}, \nu^{(k)}) \)

choose a scaling element \( p \in \mathcal{C}(s^{(k)}, \nu^{(k)}) \) and compute \( (\tilde{s}^{(k)}, \tilde{\nu}^{(k)}) \)

end for

compute \( \Delta x := -\{\nabla^2_{xx} \tilde{\eta}(\mu, x)\}^{-1} \nabla_x \tilde{\eta}(\mu, x) \Delta x \) using (3.1.19) and (3.1.20)

compute \( \delta(\mu, x) := \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x} \) using (3.1.20)

while \( \delta > \beta \) do

\[
x := x + \theta \Delta x
\]

\[
\text{for } k = 1, 2, \ldots, K \text{ do}
\]

solve (3.1.9) to obtain \( (y^{(k)}, s^{(k)}, \nu^{(k)}) \)

choose a scaling element \( p \in \mathcal{C}(s^{(k)}, \nu^{(k)}) \) and compute \( (\tilde{s}^{(k)}, \tilde{\nu}^{(k)}) \)

end for

compute \( \Delta x := -\{\nabla^2_{xx} \tilde{\eta}(\mu, x)\}^{-1} \nabla_x \tilde{\eta}(\mu, x) \Delta x \) using (3.1.19) and (3.1.20)

compute \( \delta(\mu, x) := \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x} \) using (3.1.20)

end while

\[
\mu := \gamma \mu
\]

apply inverse scaling to \( (\tilde{s}^{(k)}, \tilde{\nu}^{(k)}) \)

end while

Our algorithm is initialized with a starting point \( x^0 \in \mathcal{F}_0 \) and a starting value \( \mu^0 > 0 \) for the barrier parameter \( \mu \), and is indexed by a parameter \( \gamma \in (0, 1) \). We use \( \delta \) as a measure of the proximity of the current point \( x \) to the central path, and \( \beta \) as a threshold.
for that measure. If the current $x$ is too far away from the central path in the sense that $\delta > \beta$, Newton’s method is applied to find a point close to the central path. Then the value of $\mu$ is reduced by a factor $\gamma$ and the whole process is repeated until the value of $\mu$ is within the tolerance $\epsilon$. By tracing the central path as $\mu$ approaches zero, a strictly feasible $\epsilon$-optimal solution to (3.1.11) will be generated.

### 3.4 Complexity analysis

In this section we present the complexity analysis for two variants of algorithms: short-step algorithms and long-step algorithms, which are controlled by the input value of $\gamma$ in Algorithm 1.

As mentioned in [25], the first part of the following proposition follows directly from the definition of self concordance and is due to [30, Theorem 2.1.1]. The second part is results from the first part and is given in [48] without proof.

**Proposition 3.4.1.** For any $\mu > 0$ and $x \in F_0$, we denote $\Delta x := -\{\nabla^2_{xx} \tilde{\eta}(\mu, x)\}^{-1}\nabla_x \tilde{\eta}(\mu, x)$ and $\delta := \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x}$. Then for $\delta < 1$, $\tau \in [0, 1]$ and any $h, h_1, h_2 \in \mathbb{R}^{m_2}$ we have

\[(i)\quad -(1-\tau \delta)^2 h^T \nabla^2_{xx} \tilde{\eta}(\mu, x) h \leq -h^T \nabla^2_{xx} \tilde{\eta}(\mu, x + \tau \Delta x) h \leq -(1-\tau \delta)^{-2} h^T \nabla^2_{xx} \tilde{\eta}(\mu, x) h,
\]
\[(ii)\quad |h_1^T (\nabla^2_{xx} \tilde{\eta}(\mu, x + \tau \Delta x) - \nabla^2_{xx} \tilde{\eta}(\mu, x)) h_2| \leq [(1-\tau \delta)^{-2} - 1] \sqrt{-h_1^T \nabla^2_{xx} \tilde{\eta}(\mu, x) h_1} \sqrt{-h_2^T \nabla^2_{xx} \tilde{\eta}(\mu, x) h_2}.
\]

The following lemma is essentially Theorem 2.2.3 of [30] and describes the behavior of the Newton method as applied to $\tilde{\eta}(\mu, \cdot)$.

**Lemma 3.4.1.** For any $\mu > 0$ and $x \in F_0$, let $\Delta x$ be the Newton direction defined by $\Delta x := -\{\nabla^2_{xx} \tilde{\eta}(\mu, x)\}^{-1}\nabla_x \tilde{\eta}(\mu, x)$; $\delta := \delta(\mu, x) = \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x}$; $x^+ = x + \Delta x$, $\Delta x^+$ be the Newton direction calculated at $x^+$, and $\delta(\mu, x^+) := \sqrt{-\frac{1}{\mu} \Delta x^+^T \nabla^2_{xx} \tilde{\eta}(\mu, x^+) \Delta x^+}$. Then the following relations hold:
(i) If \( \delta < 2 - \sqrt{3} \), then \( \delta(\mu, x^+) \leq \left( \frac{\delta}{1-\delta} \right)^2 \leq \frac{\delta}{2} \).

(ii) If \( \delta \geq 2 - \sqrt{3} \), then \( \tilde{\eta}(\mu, x) - \tilde{\eta}(\mu, x + \bar{\theta} \Delta x) \geq \mu(\delta - \ln(1 + \delta)), \) where \( \bar{\theta} = (1 + \delta)^{-1} \).

### 3.4.1 Complexity for short-step algorithm

In the short-step version of the algorithm, we decrease the barrier parameter by a factor \( \gamma := 1 - \sigma/\sqrt{r_1 + Kr_2}, \) with \( \sigma < 0.1, \) in each iteration. The \( k^{th} \) iteration of the short-step algorithm is performed as follows: at the beginning of the iteration, we have \( \mu^{(k-1)} \) and \( x^{(k-1)} \) on hand and \( x^{(k-1)} \) is close to the central path, i.e., \( \delta(\mu^{(k-1)}, x^{(k-1)}) \leq \beta \).

After the barrier parameter \( \mu \) is reduced from \( \mu^{(k-1)} \) to \( \mu^k := \gamma \mu^{(k-1)} \), we have that \( \delta(\mu^k, x^{(k-1)}) \leq 2\beta \). Then a full Newton step with size \( \theta = 1 \) is taken to produce a new point \( x^k \) with \( \delta(\mu^k, x^k) \leq \beta \). We now show that, in this class of algorithms, only one Newton step is sufficient for recentering after updating the parameter \( \mu \). For the purpose of proving this result, we present the following proposition which is a restatement of [30, Theorem 3.1.1].

**Proposition 3.4.2.** Let \( \chi_\kappa(\tilde{\eta}; \mu, \mu^+) := \left( \frac{1+2r_2}{\kappa} + \frac{\sqrt{r_1 + Kr_2}}{\kappa} \right) \ln \gamma^{-1} \). Assume that \( \delta(\mu, x) < \kappa \) and \( \mu^+ := \gamma \mu \) satisfies

\[
\chi_\kappa(\tilde{\eta}; \mu, \mu^+) \leq 1 - \frac{\delta(\mu, x)}{\kappa}.
\]

Then \( \delta(\mu^+, x) < \kappa \).

**Lemma 3.4.2.** Let \( \mu^+ = \gamma \mu \), where \( \gamma = 1 - \sigma/\sqrt{r_1 + Kr_2} \) and \( \sigma \leq 0.1 \), and let \( \beta = (2 - \sqrt{3})/2 \). If \( \delta(\mu, x) \leq \beta \), then \( \delta(\mu^+, x) \leq 2\beta \).

**Proof.** Let \( \kappa := 2\beta = 2 - \sqrt{3} \). Since \( \delta(\mu, y) \leq \kappa/2 \), one can verify that for \( \sigma \leq 0.1, \) \( \mu^+ \) satisfies

\[
\chi_\kappa(\tilde{\eta}; \mu, \mu^+) \leq \frac{1}{2} \leq 1 - \frac{\delta(\mu, y)}{\kappa}.
\]

By Proposition 3.4.2, we have \( \delta(\mu^+, y) \leq \kappa \). \( \square \)
By Lemmas 3.4.1(i) and 3.4.2, we conclude that we can reduce the parameter $\mu$ by the factor $\gamma := 1 - \sigma/\sqrt{r_1 + Kr_2}, \sigma < 0.1$, at each iteration, and that only one Newton step is sufficient to restore proximity to the central path. So we have the following complexity result for short-step algorithms.

**Theorem 3.4.1.** Consider Algorithm 1 and let $\mu^0$ be the initial barrier parameter, $\epsilon > 0$ the stopping criterion, and $\beta = (2 - \sqrt{3})/2$. If the starting point $x^0$ is sufficiently close to the central path, i.e., $\delta(\mu^0, x^0) \leq \beta$, then the short-step algorithm reduces the barrier parameter $\mu$ at a linear rate and terminates with at most $O(\sqrt{r_1 + Kr_2} \ln(\mu^0/\epsilon))$ iterations.

### 3.4.2 Complexity for long-step algorithm

In the long-step version of the algorithm, the barrier parameter is decreased by an arbitrary constant factor $\gamma \in (0, 1)$. It has a potential for larger decrease on the objective function value, however, several damped Newton steps might be needed for restoring the proximity to the central path.

The $k^{th}$ iteration of the long-step algorithms is performed as follows: at the beginning of the iteration we have a point $x^{(k-1)}$, which is sufficiently close to $x(\mu^{(k-1)})$, where $x(\mu^{(k-1)})$ is the solution to (3.1.11) for $\mu := \mu^{(k-1)}$. The barrier parameter is reduced from $\mu^{(k-1)}$ to $\mu^k := \gamma \mu^{(k-1)}$, where $\gamma \in (0, 1)$, and then a search is started to find a point $x^k$ that is sufficiently close to $x(\mu^k)$. The long-step algorithm generates a finite sequence consisting of $N$ points in $F_0$, and we finally take $x^k$ to be equal to the last point of this sequence. We want to determine an upper bound on $N$, the number of Newton iterations that are needed to find the point $x^k$.

For $\mu > 0$ and $x \in F_0$, we define the function $\phi(\mu, x) := \tilde{\eta}(\mu, x(\mu)) - \tilde{\eta}(\mu, x)$, which represents the difference between the objective value $\tilde{\eta}(\mu^k, x^{(k)})$ at the end of $k^{th}$ iteration and the minimum objective value $\tilde{\eta}(\mu^k, x(\mu^{k-1}))$ at the beginning of $k^{th}$ iteration. Then
our task is to find an upper bound on $\phi(\mu^+, x)$. To do so, we first give upper bounds on $\phi(\mu, x)$ and $\phi'(\mu, x)$ respectively. We have the following lemma.

**Lemma 3.4.3.** Let $\mu > 0$ and $x \in F_0$, we denote $\tilde{\Delta}x := x(\mu) - x$ and define

$$
\tilde{\delta} := \tilde{\delta}(\mu, x) = \sqrt{-\frac{1}{\mu} \tilde{\Delta}x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \tilde{\Delta}x}.
$$

For any $\mu > 0$ and $x \in F_0$, if $\tilde{\delta} < 1$, then the following inequalities hold:

$$
\phi(\mu, x) \leq \mu \left( \frac{\tilde{\delta}}{1 - \tilde{\delta}} + \ln(1 - \tilde{\delta}) \right), \quad (3.4.1)
$$

$$
|\phi'(\mu, x)| \leq -\sqrt{r_1 + Kr_2 \ln(1 - \tilde{\delta})}. \quad (3.4.2)
$$

**Proof.**

$$
\phi(\mu, x) := \tilde{\eta}(\mu, x(\mu)) - \tilde{\eta}(\mu, x) := \int_0^1 \nabla_x \tilde{\eta}(\mu, x + \tau \tilde{\Delta}x)^T \tilde{\Delta}x d\tau.
$$

Since $x(\mu)$ is the optimal solution, we have

$$
\nabla_x \tilde{\eta}(\mu, x(\mu)) = 0 \quad (3.4.3)
$$

Hence, with the aid of Proposition 3.4.1(i), we get

$$
\phi(\mu, x) = \int_0^1 \int_0^\tau \tilde{\Delta}x^T \nabla^2_{xx} \tilde{\eta}(\mu, x + \alpha \tilde{\Delta}x) \tilde{\Delta}x \, d\alpha \, d\tau \\
\leq - \int_0^1 \int_\tau^1 \tilde{\Delta}x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \tilde{\Delta}x \, d\alpha \, d\tau \\
= \int_\tau^1 \frac{\mu \tilde{\delta}^2}{(1 - \alpha \tilde{\delta})^2} \, d\alpha \, d\tau \\
= \mu \left( \frac{\tilde{\delta}}{1 - \tilde{\delta}} + \ln(1 - \tilde{\delta}) \right),
$$

which establishes (3.4.1).

Now, for any $\mu > 0$, by applying the chain rule, using (3.4.3), and applying the
Mean-Value Theorem, we get
\[
\phi'(\mu, x) = \tilde{\eta}'(\mu, x(\mu)) - \tilde{\eta}'(\mu, x) + \nabla_x \tilde{\eta}(\mu, x(\mu))^T x'(\mu)
\]
\[
= \tilde{\eta}'(\mu, x(\mu)) - \tilde{\eta}'(\mu, x) = \nabla_x \tilde{\eta}(\mu, x + \varpi \Delta x)^T \Delta x,
\]
for some \(\varpi \in (0, 1)\). Hence
\[
|\phi'(\mu, x)| = \left| \int_0^1 \nabla_x \tilde{\eta}'(\mu, x + \tau \Delta x)^T \Delta x \, d\tau \right|
\leq \int_0^1 \sqrt{-\Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x + \tau \Delta x) \Delta x}
\sqrt{-\nabla_x \tilde{\eta}'(\mu, x + \tau \Delta x)^T [\nabla^2_{xx} \tilde{\eta}(\mu, x + \tau \Delta x)]^{-1} \nabla_x \tilde{\eta}'(\mu, x + \tau \Delta x)} \, d\tau.
\]
Then, by using (3.2.5) and Proposition 3.4.1(i), we obtain
\[
|\phi'(\mu, x)| \leq \int_0^1 \sqrt{-\Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x}
\sqrt{\frac{r_1 + Kr_2}{\mu}} \frac{1}{1 - \tau \delta} \, d\tau
\leq \int_0^1 \frac{\sqrt{r_1 + Kr_2}}{\mu} \, d\tau
= \sqrt{r_1 + Kr_2} \int_0^1 \frac{1}{1 - \tau \delta} \, d\tau
= -\sqrt{r_1 + Kr_2} \ln(1 - \tilde{\delta}),
\]
which establishes (3.4.2). \(\square\)

**Lemma 3.4.4.** Let \(\mu > 0 \) and \(x \in F_0\) be such that \(\tilde{\delta} < 1\), where \(\tilde{\delta}\) is as defined in Lemma 3.4.3. Let \(\mu^+ := \gamma \mu\) with \(\gamma \in (0, 1)\). Then
\[
\tilde{\eta}(\mu^+, x(\mu^+)) - \tilde{\eta}(\mu^+, x) \leq O(r_1 + Kr_2)\mu^+.
\]

**Proof.** Differentiating (3.4.4) with respect to \(\mu\), we get
\[
\phi''(\mu, x) = \tilde{\eta}''(\mu, x(\mu)) - \tilde{\eta}''(\mu, x) + \nabla_x \tilde{\eta}'(\mu, x(\mu))^T x'(\mu).
\]

69
We will work on the right-hand side of (3.4.5) by verifying that the second term \( \tilde{\eta}''(\mu, \mathbf{x}) \) is nonnegative and then bounding the first and the last term.

From the definition of \( \tilde{\eta}(\mathbf{x}, \mu) \), we have that \( \tilde{\eta}''(\mu, \mathbf{x}) = \sum_{k=1}^{K} \rho_k''(\mu, \mathbf{x}) \). Differentiating \( \rho_k(\mu, \mathbf{x}) \) with respect to \( \mu \) and using (3.2.4) and (3.1.10) gives

\[
\rho_k'(\mu, \mathbf{x}) = \mathbf{d}^{(k) T} \mathbf{y}^{(k)'} + \ln \det \tilde{s}^{(k)} + \mu \tilde{s}^{(k)-1} \cdot \tilde{s}^{(k)'} = \ln \det \tilde{s}^{(k)} + \mathbf{d}^{(k) T} \left( -R^{(k)-1} \tilde{W}^{(k) T} \tilde{s}^{(k)-1} \right) + \mu \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \ln \det \tilde{s}^{(k)} + \mathbf{d}^{(k) T} \left( -R^{(k)-1} \tilde{W}^{(k) T} \tilde{s}^{(k)-1} \right) + \mu \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \ln \det \tilde{s}^{(k)} + \mathbf{d}^{(k) T} \left( -R^{(k)-1} \tilde{W}^{(k) T} \tilde{s}^{(k)-1} \right) + \mu \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \ln \det \tilde{s}^{(k)}.
\]

Observe that \( (I - P^{(k)})^2 = I - P^{(k)} \) (remember \( P^{(k)2} = P^{(k)} \)). Therefore, by differentiating \( \rho_k'(\mu, \mathbf{x}) \) with respect to \( \mu \) and using (3.2.4) and (3.1.18), we obtain

\[
\rho_k''(\mu, \mathbf{x}) = \tilde{s}^{(k)-1} \cdot \tilde{s}^{(k)'} = \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
\geq 0.
\]

Thus, for \( \mu > 0 \) and \( \mathbf{x} \in \mathcal{F}_0 \), \( \tilde{\eta}''(\mu, \mathbf{x}) \geq 0 \), and hence \( \tilde{\eta}(\mu, \mathbf{x}) \) is a convex function of \( \mu \). In addition, using (3.1.18), we also have

\[
\rho_k''(\mu, \mathbf{x}) = \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
= \tilde{s}^{(k)-1} \cdot \left( \tilde{W}^{(k)} R^{(k)-1} \tilde{W}^{(k)} \tilde{s}^{(k)-1} \right)
\]

\[
\leq \tilde{s}^{(k)-1} \cdot \tilde{s}^{(k)'}\]
\[
\begin{align*}
\mu^{-1} \tilde{\nu}^{(k)} \bullet \tilde{s}^{(k)} \\
= \mu^{-1} \text{trace}(e_2) \\
= \frac{r_2}{\mu}.
\end{align*}
\]

Hence

\[
\tilde{\eta}''(\mu, x(\mu)) \leq \frac{Kr_2}{\mu}.
\] (3.4.6)

By differentiating (3.4.3) with respect to \( \mu \), we get

\[
\nabla_x \tilde{\eta}'(\mu, x(\mu)) + \nabla^2_{xx} \tilde{\eta}(\mu, x(\mu)) x'(\mu) = 0,
\]

or equivalently

\[
x'(\mu) = -\left\{ \nabla^2_{xx} \tilde{\eta}(\mu, x(\mu)) \right\}^{-1} \nabla_x \tilde{\eta}'(\mu, x(\mu)).
\]

Hence, by using (3.2.5), we have

\[
-\nabla_x \tilde{\eta}'(\mu, x(\mu))^T x'(\mu) = \nabla_x \tilde{\eta}'(\mu, x(\mu))^T \left\{ \nabla^2_{xx} \tilde{\eta}(\mu, x(\mu)) \right\}^{-1} \nabla_x \tilde{\eta}'(\mu, x(\mu))
\leq \frac{1}{\mu} (r_1 + Kr_2).
\] (3.4.7)

By combining (3.4.6) and (3.4.7), and using the fact that \( \tilde{\eta}''(\mu, x) \geq 0 \), we obtain

\[
\phi''(\mu, x(\mu)) \leq \frac{r_1 + 2Kr_2}{\mu}.
\] (3.4.8)

Applying the Mean-Value Theorem and using Lemma 3.4.3 and (3.4.8) gives

\[
\begin{align*}
\phi(\mu^+, x) & = \phi(\mu, x) + \phi'(\mu, x)(\mu^+ - \mu) + \int_{\mu}^{\mu^+} \int_{\mu}^{\tau} \phi''(\nu, x) \, d\nu d\tau \\
& \leq \mu \left( \frac{\delta}{1 - \delta} + \ln(1 - \tilde{\delta}) \right) - \sqrt{r_1 + Kr_2} \ln(1 - \tilde{\delta}) (\mu - \mu^+) \\
& \quad + (r_1 + 2Kr_2) \int_{\mu}^{\mu^+} \int_{\mu}^{\tau} \nu^{-1} \, d\nu d\tau.
\end{align*}
\]

71
\[
= \mu \left( \frac{\tilde{\delta}}{1-\delta} + \ln(1-\tilde{\delta}) \right) - \sqrt{r_1 + K r_2} \ln(1-\tilde{\delta}) \ (\mu - \mu^+)
\]
\[
+(r_1 + 2K r_2) \ (\mu - \mu^+) \ln \frac{\xi}{\mu}
\]
\[
\leq \mu \left( \frac{\tilde{\delta}}{1-\delta} + \ln(1-\tilde{\delta}) \right) - \sqrt{r_1 + K r_2} \ln(1-\tilde{\delta}) \ (\mu - \mu^+)
\]
\[
+(r_1 + 2K r_2) \ (\mu - \mu^+) \ln \gamma^{-1}
\]

(recall $\gamma^{-1} = \mu^+ / \mu \geq \tau / \mu$). Since $\tilde{\delta}$ and $\gamma$ are constants, the lemma is established. \(\square\)

Notice that the previous lemma requires $\tilde{\delta} < 1$. However, evaluating $\tilde{\delta}$ explicitly may not be possible. In the next lemma we will see that $\tilde{\delta}$ is actually proportional to $\delta$, which can be evaluated.

**Lemma 3.4.5.** For any $\mu > 0$ and $x \in F_0$, let $\Delta x := -\left(\nabla^2_{xx} \tilde{\eta}(\mu, x)\right)^{-1} \nabla_x \tilde{\eta}(\mu, x)$ and $\tilde{\Delta} x := x - x(\mu)$. We denote

\[
\delta := \delta(\mu, x) = \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x} \quad \text{and} \quad \tilde{\delta} := \tilde{\delta}(\mu, x) = \sqrt{-\frac{1}{\mu} \tilde{\Delta} x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \tilde{\Delta} x}.
\]

If $\delta < 1/6$, then $\frac{2}{3} \delta \leq \tilde{\delta} \leq 2 \delta$.

**Proof.** Let $H := \nabla^2_{xx} \tilde{\eta}(\mu, x)$ and $g := \nabla_x \tilde{\eta}(\mu, x)$, and denote $\tilde{g} := g + H \tilde{\Delta} x$. From (3.4.3), we have $\Delta x = -H^{-1} g$. This gives us $\tilde{\Delta} x = \Delta x + H^{-1} \tilde{g}$. We then have

\[
\tilde{\delta} = \sqrt{-\frac{1}{\mu} (\Delta x + H^{-1} \tilde{g})^T \nabla^2_{xx} \tilde{\eta}(\mu, x) (\Delta x + H^{-1} \tilde{g})}
\]
\[
\leq \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x - \frac{1}{\mu} (H^{-1} \tilde{g})^T \nabla^2_{xx} \tilde{\eta}(\mu, x) (H^{-1} \tilde{g})}
\]
\[
\leq \delta + \sqrt{-\frac{1}{\mu} \tilde{g}^T H^{-1} \tilde{g}},
\]

where we used the triangle inequality to obtain the last inequality. Note that, by (3.4.3),

72
we have $\nabla_x \tilde{\eta}(\mu, x - \tilde{\Delta}x) = 0$. Applying the Mean-Value Theorem gives

\[
-h^T \bar{g} = -h^T (H \tilde{\Delta}x + g) \\
= -h^T (\nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x + \nabla_x \tilde{\eta}(\mu, x)) \\
= -h^T (\nabla^2_{xx} \tilde{\eta}(\mu, x) \Delta x - (\nabla_x \tilde{\eta}(\mu, x - \Delta x) - \nabla_x \tilde{\eta}(\mu, x))) \\
= -h^T (\nabla^2_{xx} \tilde{\eta}(\mu, x) - \nabla^2_{xx} \tilde{\eta}(\mu, x - (1 - \varpi) \Delta x)) \tilde{\Delta}x,
\]

for some $\varpi \in (0, 1)$.

Now in view of Proposition 3.4.1(ii) we have

\[
-h^T \bar{g} = -\int_0^1 h^T (\nabla^2_{xx} \tilde{\eta}(\mu, x) - \nabla^2_{xx} \tilde{\eta}(\mu, x - (1 - \tau) \Delta x)) \tilde{\Delta}x \, d\tau \\
\leq \sqrt{-\Delta x^T H \Delta x} \sqrt{-h^T H h} \int_0^1 ((1 - (1 - \tau) \tilde{\delta})^{-2} - 1) \, d\tau \\
= \left( \frac{\tilde{\delta}}{1 - \tilde{\delta}} \right) \sqrt{-\Delta x^T H \Delta x} \sqrt{-h^T H h} \\
= \left( \frac{\sqrt{\mu} \tilde{\delta}^2}{1 - \tilde{\delta}} \right) \sqrt{-h^T H h}.
\]

It can be verified that $-\bar{g}^T H^{-1} \bar{g} = \max \{ h^T H h - 2h^T \bar{g} : h \in \mathbb{R}^{m_1} \}$. It then follows that

\[
-\bar{g}^T H^{-1} \bar{g} \leq \max \left\{ h^T H h + \left( \frac{2\sqrt{\mu} \tilde{\delta}^2}{1 - \tilde{\delta}} \right) \sqrt{-h^T H h} : h \in \mathbb{R}^{m_1} \right\} = \frac{\mu \tilde{\delta}^4}{(1 - \tilde{\delta})^2}. \tag{3.4.10}
\]

From (3.4.9) and (3.4.10), we obtain $\tilde{\delta} \leq \delta + \frac{\tilde{\delta}^2}{1 - \tilde{\delta}}$, or equivalently, $\tilde{\delta}^2 - (1 + \delta) \tilde{\delta} + \delta \geq 0$. Therefore, the condition $\delta \leq 1/6$ implies that $12 \tilde{\delta}^2 - 7 \tilde{\delta} + 1 \geq 0$, which in turn gives that $\tilde{\delta} \leq 1/3 = 2\delta$.

From (3.4.9), by exchanging positions of $\Delta x$ and $\tilde{\Delta} x$ and following the above steps, we get

\[
\delta \leq \tilde{\delta} + \frac{\tilde{\delta}^2}{1 - \tilde{\delta}}, \text{ or equivalently, } \delta \leq \frac{\tilde{\delta}}{1 - \tilde{\delta}} \leq \frac{\tilde{\delta}}{1 - \frac{1}{3}} = \frac{3}{2} \tilde{\delta}.
\]

Thus, the condition $\delta < 1/6$ implies that $\frac{3}{2} \delta \leq \tilde{\delta} \leq 2\delta$. This completes the proof. □
Combining Lemmas 3.4.1(ii), 3.4.4, and 3.4.5, we have the following complexity result for long-step algorithms.

**Theorem 3.4.2.** Consider Algorithm 1 and let $\mu^0$ be the initial barrier parameter, $\epsilon > 0$ the stopping criterion, and $\beta = 1/6$. If the starting point $x^0$ is sufficiently close to the central path, i.e., $\delta(\mu^0, x^0) \leq \beta$, then the long-step algorithm reduces the barrier parameter $\mu$ at a linear rate and terminates with at most $O((r_1 + Kr_2) \ln(\mu^0/\epsilon))$ iterations.
Chapter 4

A Class of Polynomial Volumetric Barrier Decomposition Algorithms for Stochastic Symmetric Programming

Ariyawansa and Zhu [10] have derived a class of polynomial time decomposition-based algorithms for solving the SSDP problem (Problem 4) based on a volumetric barrier analogous to work of Mehrotra and Özevin [25] by utilizing the work of Anstreicher [7] for DSDP. In this chapter, we extend Ariyawansa and Zhu’s work [10] to the case of SSPs by deriving a class of volumetric barrier decomposition algorithms for the general SSP problem and establishing polynomial complexity of certain members of the class of algorithms. The results of this chapter have been submitted for publication [4].
4.1 The volumetric barrier problem for SSPs

In this section we formulate appropriate volumetric barrier function for the SSP problem (with finite event space Ω) and obtain expressions for the derivatives required in the rest of the paper. Our procedure closely follows that in §3 of [10], although our setting is much more general.

4.1.1 Formulation and assumptions

We now examine (2.1.3, 2.1.4) when the event space Ω is finite. Let \( \{(T^{(k)}, W^{(k)}, h^{(k)}, d^{(k)}) : k = 1, 2, \ldots, K\} \) be the set of the possible values of the random variables \( (T(\omega), W(\omega), h(\omega), d(\omega)) \) and let \( p_k := P(T(\omega), W(\omega), h(\omega), d(\omega)) = (T^{(k)}, W^{(k)}, h^{(k)}, d^{(k)}) \) be the associated probability for \( k = 1, 2, \ldots, K \). Then Problem (2.1.3, 2.1.4) becomes

\[
\max c^T x + \sum_{k=1}^{K} p_k Q^{(k)}(x) \\
\text{s.t. } A x \preceq_{K \cdot J_1} b,
\]

where, for \( k = 1, 2, \ldots, K \), \( Q^{(k)}(x) \) is the maximum of the problem

\[
\max d^{(k)T} y^{(k)} \\
\text{s.t. } W^{(k)} y^{(k)} \preceq_{K \cdot J_2} h^{(k)} - T^{(k)} x,
\]

where \( x \in \mathbb{R}^{m_1} \) is the first-stage decision variable, and \( y^{(k)} \in \mathbb{R}^{m_2} \) is the second-stage variable for \( k = 1, 2, \ldots, K \).

We notice that the constraints in (4.1.1, 4.1.2) are negative symmetric while the common practice in the DSP literature is to use positive symmetric constraints. So for convenience we redefine \( d^{(k)} \) as \( d^{(k)} := p_k d^{(k)} \) for \( k = 1, 2, \ldots, K \), and rewrite Problem (4.1.1,
4.1.2) as

$$
\begin{align*}
\min_{x} & \quad c^T x + \sum_{k=1}^{K} Q^{(k)}(x) \\
\text{s.t.} & \quad A x - b \succeq_{K, \mathcal{J}_1} 0,
\end{align*}
$$

(4.1.3)

where, for $k = 1, 2, \ldots, K$, $Q^{(k)}(x)$ is the maximum of the problem

$$
\begin{align*}
\min_{y^{(k)}} & \quad d^{(k)^T} y^{(k)} \\
\text{s.t.} & \quad W^{(k)} y^{(k)} + T^{(k)} x - h^{(k)} \succeq_{K, \mathcal{J}_2} 0.
\end{align*}
$$

(4.1.4)

In the rest of this paper our attention will be on Problem (4.1.3, 4.1.4), and from now on when we use the acronym “SSP” in this paper we mean Problem (4.1.3, 4.1.4).

### 4.1.2 The volumetric barrier problem for SSPs

In this section we formulate a volumetric barrier for SSPs and obtain expressions for the derivatives required in our subsequent development.

In order to define the volumetric barrier problem for the SSP (4.1.3, 4.1.4), we need to make some assumptions. First we define

$$
\begin{align*}
\mathcal{F}_1 & := \{ x : s_1(x) := Ax - b \succ_{K, \mathcal{J}_1} 0 \}; \\
\mathcal{F}^{(k)}(x) & := \{ y^{(k)} : s^{(k)}_2(x, y^{(k)}) := W^{(k)} y^{(k)} + T^{(k)} x - h^{(k)} \succ_{K, \mathcal{J}_2} 0 \} \text{ for } k = 1, 2, \ldots, K; \\
\mathcal{F}_2 & := \{ x : \mathcal{F}^{(k)}(x) \neq \emptyset, k = 1, 2, \ldots, K \}; \\
\mathcal{F}_0 & := \mathcal{F}_1 \cap \mathcal{F}_2.
\end{align*}
$$

Now we make

**Assumption 4.1.1.** The matrix $A$ and every matrix $T^{(k)}$ have full column rank.

**Assumption 4.1.2.** The set $\mathcal{F}_0$ is nonempty.

**Assumption 4.1.3.** For each $x \in \mathcal{F}_0$ and for $k = 1, 2, \ldots, K$, Problem (4.1.4) has a nonempty isolated compact set of minimizers.
Assumption 4.1.1 is for convenience. Under Assumption 4.1.2, the set $\mathcal{F}_1$ is nonempty. The logarithmic barrier [30] for $\mathcal{F}_1$ is the function $l_1 : \mathcal{F}_1 \to \mathbb{R}$ defined by

$$l_1(x) := -\ln \det(s_1(x)), \quad \forall x \in \mathcal{F}_1,$$

and the volumetric barrier [30, 40] for $\mathcal{F}_1$ is the function $v_1 : \mathcal{F}_1 \to \mathbb{R}$ defined by

$$v_1(x) := \frac{1}{2} \ln \det(\nabla^2_{xx}l_1(x)), \quad \forall x \in \mathcal{F}_1.$$

Also under Assumption 4.1.2, $\mathcal{F}_2$ is nonempty and for $x \in \mathcal{F}_2$, $\mathcal{F}^{(k)}(x)$ is nonempty for $k = 1, 2, \ldots, K$. The logarithmic barrier [30] for $\mathcal{F}^{(k)}(x)$ is the function $l_2^{(k)} : \mathcal{F}_2 \times \mathcal{F}^{(k)}(x) \to \mathbb{R}$ defined by

$$l_2^{(k)}(x, y^{(k)}) := -\ln \det(s_2^{(k)}(x, y^{(k)})), \quad \forall y^{(k)} \in \mathcal{F}^{(k)}(x), x \in \mathcal{F}_2,$$

and the volumetric barrier [30, 40] for $\mathcal{F}^{(k)}(x)$ is the function $v_2^{(k)} : \mathcal{F}_2 \times \mathcal{F}^{(k)}(x) \to \mathbb{R}$ defined by

$$v_2^{(k)}(x, y^{(k)}) := \frac{1}{2} \ln \det(\nabla^2_{yy}l_2^{(k)}(x, y^{(k)})), \quad \forall y^{(k)} \in \mathcal{F}^{(k)}(x), x \in \mathcal{F}_2.$$

Now, we define the volumetric barrier problem for the SSP (4.1.3, 4.1.4) as

$$\min \eta(\mu, x) : = c^T x + \sum_{k=1}^K \rho_k(\mu, x) + \mu c_1 v_1(x), \quad (4.1.5)$$

where for $k = 1, 2, \ldots, K$ and $x \in \mathcal{F}_0$, $\rho_k(\mu, x)$ is the minimum of the problem

$$\min d^{(k)^T} y^{(k)} + \mu c_2 v_2^{(k)}(x, y^{(k)}), \quad (4.1.6)$$

78
Here \( c_1 := 225\sqrt{n_1} \) and \( c_2 := 450n_2^3 \) are constants, and \( \mu > 0 \) is the barrier parameter.

Now, we will show that (4.1.6) has a unique minimizer for each \( x \in \mathcal{F}_0 \) and for \( k = 1, 2, \ldots, K \). For this purpose, we present the following theorem.

**Theorem 4.1.1** (Fiacco and Mccormick [19, Theorem 8]). Consider the inequality constrained problem

\[
\min f(x) \quad \text{s.t.} \quad g_i(x) \geq 0, \quad i = 1, 2, \ldots, m,
\]

where the functions \( f, g_1, \ldots, g_m : \mathbb{R}^n \to \mathbb{R} \) are continuous. Let \( I \) be a scalar-valued function of \( x \) with the following two properties: \( I(x) \) is continuous in the region \( R^0 := \{ x : g_i(x) > 0, i = 1, 2, \ldots, m \} \), which is assumed to be nonempty; if \( \{x_k\} \) is any infinite sequence of points in \( R^0 \) converging to \( x_B \) such that \( g_i(x_B) = 0 \) for at least one \( i \), then \( \lim_{k \to \infty} I(x_k) = +\infty \). Let \( \tau \) be a scalar-valued function of the single variable \( s \) with the following two properties: if \( s_1 > s_2 > 0 \), then \( \tau(s_1) > \tau(s_2) > 0 \); if \( \{s_k\} \) is an infinite sequence of points such that \( \lim_{k \to \infty} s_k = 0 \), then \( \lim_{k \to \infty} \tau(s_k) = 0 \). Let \( U : R^0 \times \mathbb{R}^+ \to \mathbb{R} \) be defined by \( U(x, s) := f(x) + \tau(s)I(x) \). If (4.1.7) has a nonempty, isolated compact set of local minimizers and \( \{s_k\} \) is a strictly decreasing infinite sequence, then the unconstrained set of local minimizers of \( U(\cdot, s_k) \) exist for \( s_k \) small.

**Lemma 4.1.1.** If Assumptions 4.1.2 and 4.1.3 hold, then for each \( x \in \mathcal{F}_0 \) and \( k = 1, 2, \ldots, K \), the Problem (4.1.6) has a unique minimizer for \( \mu \) small.

**Proof.** For any \( x \in \mathcal{F}_0 \), \( v_2^{(k)}(x, y^{(k)}) \) is defined on the nonempty set \( \mathcal{F}^{(k)}(x) \). By Theorem 1.3.1, there exist real numbers \( \lambda_1^{(k)}, \lambda_2^{(k)}, \ldots, \lambda_{r_2}^{(k)} \) and a Jordan frame \( c_1^{(k)}, c_2^{(k)}, \ldots, c_{r_2}^{(k)} \) such that \( s_2^{(k)}(x, y^{(k)}) = \lambda_1^{(k)}c_1^{(k)} + \lambda_2^{(k)}c_2^{(k)} + \cdots + \lambda_{r_2}^{(k)}c_{r_2}^{(k)} \), and \( \lambda_1^{(k)}, \lambda_2^{(k)}, \ldots, \lambda_{r_2}^{(k)} \) are the eigenvalues of \( s_2^{(k)}(x, y^{(k)}) \). Moreover, \( \lambda_j^{(k)} \) can be viewed as a function of \( y^{(k)} \in \mathcal{F}^{(k)}(x) \) for \( j = 1, 2, \ldots, r_2 \). Then \( \lambda_j^{(k)} \) is continuous for \( j = 1, 2, \ldots, r_2 \) and hence the constraint \( s_2^{(k)}(x, y^{(k)}) \succ k_{J_2} 0 \) can be replaced by the constraints: \( \lambda_j^{(k)}(y^{(k)}) > 0, \ j = 1, 2, \ldots, r_2 \).

So (4.1.4) can be rewritten in the form of (4.1.7). Therefore, by Theorem 4.1.1, local
minimizers of (4.1.6) exist for each \( x \in \mathcal{F}_0 \) and \( k = 1, 2, \ldots, K \) for \( \mu \) small. The uniqueness of the minimizer follows from the fact that \( v_2^{(k)} \) is strictly convex. \( \square \)

In view of Lemma 4.1.1, Problem (4.1.5) is well-defined, and its feasible set is \( \mathcal{F}_0 \).

### 4.1.3 Computation of \( \nabla_x \eta(\mu, x) \) and \( \nabla^2_{xx} \eta(\mu, x) \)

In order to compute the derivatives of \( \eta \) we need to determine the derivatives of \( \rho_k \), \( k = 1, 2, \ldots, K \), which in turn require the derivatives of \( v_2^{(k)} \) and \( l_2^{(k)} \) for \( k = 1, 2, \ldots, K \). We will drop the superscript \( (k) \) when it does not lead to confusion.

Note that

\[
\nabla_x s_1(x) = A, \quad \nabla_x s_2(x, y) = T, \quad \text{and} \quad \nabla_y s_2(x, y) = W.
\]

Hence

\[
\nabla_x l_1(x) = -(\nabla_x s_1)^T s_1^{-1} = -A^T s_1^{-1}, \quad \text{and} \quad \nabla_y l_2(x, y) = -(\nabla_y s_2)^T s_2^{-1} = -W^T s_2^{-1}.
\]

This implies that

\[
\nabla^2_{xx} l_1(x) = A^T [s_1^{-1}] \nabla_x s_1 = A^T [s_1^{-1}] A,
\]

and

\[
H := \nabla^2_{yy} l_2(x, y) = W^T [s_2^{-1}] \nabla_x s_2 = W^T [s_2^{-1}] W.
\]

We need the matrix calculus result for our computation.

**Proposition 4.1.1.** Let \( X \in \mathbb{R}^{n \times n} \) be nonsingular. Then

\[
\frac{\partial}{\partial X_{ij}} X^{-1} = -X^{-1} e_i e_j^T X^{-1},
\]

for \( i, j = 1, 2, \ldots, n \). Here, \( e_i \) is the \( i \)th vector in the standard basis for \( \mathbb{R}^n \).
To compute the first partial derivatives of $v_2(x, y)$, we start by observing that

$$\frac{\partial}{\partial x_k} [s_2^{-1}] = \sum_{ij} \frac{\partial}{\partial [s_2]_{ij}} [s_2^{-1}] \frac{\partial}{\partial x_k} [s_2]_{ij}$$
$$= \sum_{ij} \frac{\partial}{\partial [s_2]_{ij}} [s_2]^{-1} \frac{\partial}{\partial x_k} [s_2]_{ij}$$
$$= -2 \sum_{ij} [s_2]^{-1} e_i e_j^T [s_2]^{-1} [s_2, \frac{\partial}{\partial x_k} s_2]_{ij}$$
$$= -2 \sum_{ij} [s_2]^{-1} e_i e_j^T [s_2]^{-1} [s_2, t_k]_{ij}$$
$$= -2 [s_2^{-1}] [s_2, t_k] [s_2^{-1}]. \tag{4.1.8}$$

Then, for $i = 1, 2, \ldots, m_1$, we have

$$\frac{\partial}{\partial x_i} v_2(x, y) = \frac{1}{2} \frac{\partial}{\partial x_i} \ln \det H$$
$$= \frac{1}{2} H^{-1} \cdot \frac{\partial}{\partial x_i} H$$
$$= \frac{1}{2} H^{-1} \cdot \frac{\partial}{\partial x_i} (W^T [s_2^{-1}] W)$$
$$= \frac{1}{2} H^{-1} \cdot W^T \left( \frac{\partial}{\partial x_i} [s_2^{-1}] \right) W$$
$$= -H^{-1} \cdot W^T [s_2]^{-1} [s_2, t_i] [s_2]^{-1} W$$
$$= -W^T H^{-1} W \cdot [s_2]^{-1} [s_2, t_i] [s_2]^{-1}$$
$$= -[s_2^{-1/2}] W^T H^{-1} W [s_2^{-1/2}] \cdot [s_2^{-1/2}] [s_2, t_i] [s_2^{-1/2}]$$

By defining

$$P := [s_2^{-1/2}] W^T H^{-1} W [s_2^{-1/2}],$$

which acts as the orthogonal projection onto the range of $[s_2^{-1/2}] W$, we get

$$\frac{\partial}{\partial x_i} v_2(x, y) = -W^T H^{-1} W \cdot [s_2^{-1}] [s_2, t_i] [s_2^{-1}] = -P \cdot [s_2^{-1/2}] [s_2, t_i] [s_2^{-1/2}], \tag{4.1.9}$$

for $i = 1, 2, \ldots, m_1$. 

81
Similarly, we have

$$\frac{\partial}{\partial y_i} v_2(x, y) = -W^T H^{-1} W \cdot [s_2^{-1}] [s_2, w_i] [s_2^{-1}] = -P \cdot [s_2^{-1/2}] [s_2, w_i] [s_2^{-1/2}],$$

for \( i = 1, 2, \ldots, m_2. \)

To compute the second partial derivatives of \( v_2(x, y) \), we start by observing that

$$\frac{\partial}{\partial x_k} H^{-1} = \sum_{ij} \frac{\partial}{\partial H_{ij}} H^{-1} \frac{\partial}{\partial x_k} H_{ij}$$

$$= -\sum_{ij} (H^{-1} e_i e_j^T H^{-1}) \frac{\partial}{\partial x_k} \left[ W^T [s_2^{-1}] W \right]_{ij}$$

$$= -\sum_{ij} (H^{-1} e_i e_j^T H^{-1}) \left[ W^T \left( \frac{\partial}{\partial x_k} [s_2^{-1}] \right) W \right]_{ij}$$

$$= 2 \sum_{ij} (H^{-1} e_i e_j^T H^{-1}) \left[ W^T [s_2]^{-1} [s_2, t_k] [s_2]^{-1} W \right]_{ij}$$

$$= 2 H^{-1} W^T [s_2]^{-1} [s_2, t_k] [s_2]^{-1} WH^{-1},$$

and, using (4.1.8) and Lemma 1.3.2, that

$$\frac{\partial}{\partial x_j} \left( [s_2^{-1}] [s_2, t_i] [s_2^{-1}] \right) = \left( \frac{\partial}{\partial x_j} [s_2^{-1}] \right) [s_2, t_i] [s_2^{-1}] + [s_2^{-1}] \left( \frac{\partial}{\partial x_j} [s_2, t_i] \right) [s_2^{-1}] + [s_2^{-1}] [s_2, t_i] \left( \frac{\partial}{\partial x_j} [s_2^{-1}] \right)$$

$$= -2 [s_2^{-1}] [s_2, t_i] [s_2^{-1}] [s_2, t_i] [s_2^{-1}] + [s_2^{-1}] [t_i, t_i] [s_2^{-1}]$$

$$= -2 [s_2^{-1}] [s_2, t_i] [s_2^{-1}] [s_2, t_i] [s_2^{-1}]$$

$$= -2 \left. [s_2^{-1}] [s_2, t_i] [s_2^{-1}] [s_2, t_i] [s_2^{-1}] \right)$$

$$+ \left. [s_2^{-1}] \left( [t_i, t_i] - 2 [s_2, t_i] [s_2^{-1}] [s_2, t_i] \right) [s_2^{-1}] \right).$$

By combining (4.1.9), (4.1.10) and (4.1.11), we get

$$\nabla_{xx}^2 v_2(y, x) = \frac{\partial^2}{\partial x \partial x} v_2(x, y) = 2Q^{xx} + R^{xx} - 2T^{xx},$$

(4.1.12)
where

\[ Q^{xx}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_j, t_j] [s_i^{-1}] [s_i, t_j] [s_i^{-1}] , \]
\[ R^{xx}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] \left( 2 [s_i, t_i] [s_i^{-1}] [s_i, t_i] - [t_j, t_i] \right) [s_i^{-1}] , \]
\[ T^{xx}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_i, t_i] [s_i^{-1}] WH^{-1}W [s_i^{-1}] [s_i, t_i] [s_i^{-1}] . \]

By following similar steps as above, we have that

\[ \nabla^2_{yy} v_2(y, x) = \frac{\partial^2}{\partial y \partial y} v_2(x, y) = 2Q^{yy} + R^{yy} - 2T^{yy}, \]

where

\[ Q^{yy}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_j, w_j] [s_i^{-1}] [s_i, w_i] [s_i^{-1}] , \]
\[ R^{yy}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] \left( 2 [s_i, w_i] [s_i^{-1}] [s_i, w_j] - [w_j, w_i] \right) [s_i^{-1}] , \]
\[ T^{yy}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_i, w_j] [s_i^{-1}] WH^{-1}W [s_i^{-1}] [s_i, w_i] [s_i^{-1}] ; \]

\[ \nabla^2_{xy} v_2(x, y) = \frac{\partial^2}{\partial y \partial x} v_2(x, y) = 2Q^{xy} + R^{xy} - 2T^{xy}, \]

where

\[ Q^{xy}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_j, w_j] [s_i^{-1}] [s_i, t_i] [s_i^{-1}] , \]
\[ R^{xy}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] \left( 2 [s_i, t_i] [s_i^{-1}] [s_i, w_j] - [w_j, t_i] \right) [s_i^{-1}] , \]
\[ T^{xy}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_i, w_j] [s_i^{-1}] WH^{-1}W [s_i^{-1}] [s_i, t_i] [s_i^{-1}] ; \]

and

\[ \nabla^2_{yx} v_2(y, x) = \frac{\partial^2}{\partial x \partial y} v_2(x, y) = 2Q^{yx} + R^{yx} - 2T^{yx}, \]

where

\[ Q^{yx}_{i,j} = (WH^{-1}W^T) \cdot [s_i^{-1}] [s_j, t_j] [s_i^{-1}] [s_i, w_i] [s_i^{-1}] , \]
\[ R_{ij} = (WH^{-1}W^T) \cdot [s_2^{-1} \left( 2 [s_2, w_i] [s_2^{-1}] [s_2, t_j] - [t_j, w_i] \right) [s_2^{-1}], \]
\[ T_{yx} = (WH^{-1}W^T) \cdot [s_2^{-1} [s_2, t_j] [s_2^{-1}] W H^{-1} W [s_2^{-1}] [s_2, w_i] [s_2^{-1}]. \]

Now, define \( \varphi_k : \mathbb{R}^+ \times F_0 \times F^{(k)}(x) \rightarrow \mathbb{R} \) by

\[ \varphi_k(\mu, x, y) := d^Ty + \mu c_2 v_2(x, y). \]

By (4.1.6) we then have

\[ \rho_k(\mu, x) = \min_{y \in F^{(k)}(x)} \varphi_k(\mu, x, y) \]

and

\[ \rho_k(\mu, x) = \varphi_k(\mu, x, y)|_{y=\bar{y}} = \varphi_k(\mu, x, \bar{y}), \]

where \( \bar{y} \) is the minimizer of (4.1.6). Observe that \( \bar{y} \) is a function of \( x \) and is defined by

\[ \nabla_y \varphi_k(\mu, x, y)|_{y=\bar{y}} = 0. \]  \hspace{1cm} (4.1.13)

Note that, by (4.1.13), we have \( \nabla_{\bar{y}} v_2(x, \bar{y}) = \nabla_y v_2(x, y)|_{y=\bar{y}} = -\frac{1}{\mu c_2} d \). This implies that

\[ \nabla^2_{\bar{y}x} v_2(x, \bar{y}) = \nabla_{x} \nabla_{\bar{y}} v_2(x, \bar{y}) = 0 \text{ and } \nabla^3_{\bar{y}xx} v_2(x, \bar{y}) = \nabla_{xx} \nabla_{y} v_2(x, \bar{y}) = 0. \]

Now we are ready to calculate the first and second order derivatives of \( \rho_k \) with respect to \( x \). We have

\[
\nabla_x \rho_k(\mu, x) = [\nabla_x \varphi_k(\mu, x, y) + \nabla_y \varphi_k(\mu, x, y) \nabla_x y]|_{y=\bar{y}} \\
= \nabla_x \varphi_k(\mu, x, y)|_{y=\bar{y}} + \nabla_y \varphi_k(\mu, x, y)|_{y=\bar{y}} \nabla_x y|_{y=\bar{y}} \\
= \nabla_x \varphi_k(\mu, x, y)|_{y=\bar{y}} \\
= \mu c_2 \nabla_x v_2(x, y)|_{y=\bar{y}}
\]
\[

\begin{align*}
\nabla^2_{xx}\rho_k(\mu, \bar{x}) &= \mu \nabla \{ \nabla_{x} v_2(\bar{x}, \bar{y}) \} \\
&= \mu c_2 \{ \nabla^2_{xx} v_2(\bar{x}, \bar{y}) + \nabla^2_{x} v_2(\bar{x}, \bar{y}) [\nabla_{x} y]_{y=\bar{y}} \} \\
&= \mu c_2 \nabla^2_{xx} v_2(\bar{x}, \bar{y}),
\end{align*}
\]

\[

\begin{align*}
\nabla^3_{xxx}\rho_k(\mu, \bar{x}) &= \mu \nabla \{ \nabla^2_{xx} v_2(\bar{x}, \bar{y}) \} \\
&= \mu c_2 \{ \nabla^3_{xxx} v_2(\bar{x}, \bar{y}) + \nabla^3_{x} v_2(\bar{x}, \bar{y}) [\nabla_{x} y]_{y=\bar{y}} \} \\
&= \mu c_2 \nabla^3_{xxx} v_2(\bar{x}, \bar{y}).
\end{align*}
\]

In summary we have

\[
\begin{align*}
\nabla_{x} \rho_k(\mu, \bar{x}) &= \mu c_2 \nabla_{x} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}), \\
\nabla^2_{xx} \rho_k(\mu, \bar{x}) &= \mu c_2 \nabla^2_{xx} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}), \\
\nabla^3_{xxx} \rho_k(\mu, \bar{x}) &= \mu c_2 \nabla^3_{xxx} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}),
\end{align*}
\]

and

\[
\begin{align*}
\nabla_{x} \eta(\mu, \bar{x}) &= c + \mu c_1 \nabla_{x} v_1(\bar{x}) + \sum_{k=1}^{K} \mu c_2 \nabla_{x} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}), \\
\nabla^2_{xx} \eta(\mu, \bar{x}) &= \mu c_1 \nabla^2_{xx} v_1(\bar{x}) + \sum_{k=1}^{K} \mu c_2 \nabla^2_{xx} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}),
\end{align*}
\]

where \( \nabla_{x} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}) \), \( \nabla^2_{xx} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}) \), and \( \nabla^3_{xxx} v_2^{(k)}(\bar{x}, \bar{y}^{(k)}) \) are calculated in (4.1.9), (4.1.12), and (4.2.4) respectively.

### 4.2 Self-concordance properties of the volumetric barrier recourse

In this section we prove that the recourse function with volumetric barrier is a strongly self-concordant function leading to a strongly self-concordant family with appropriate
parameters. Establishing this allows us to develop volumetric barrier polynomial time path following interior point methods for solving SSPs.

We need the following proposition for proving some results.

**Proposition 4.2.1.** Let $A, B, C \in \mathbb{R}^{n \times n}$. Then

1. $A, B \succeq 0$ implies that $A \cdot B \geq 0$;

2. if $A \succeq 0$ and $B \succeq C$, then $A \cdot B \geq A \cdot C$.

### 4.2.1 Self-Concordance of $\eta(\mu, \cdot)$

This subsection is devoted to show that $\eta(\mu, \cdot)$ is a strongly self-concordant barrier on $\mathcal{F}_0$ (see Definition 3.2.1). Throughout this subsection, with respect to $h \in \mathbb{R}^{m_1}$, we define

$$b := b(h) := \sum_{i=1}^{m_1} h_i t_i \quad \text{and} \quad \bar{b} := \left[ s_2^{-1/2} \begin{bmatrix} s_2, b \end{bmatrix} s_2^{-1/2} \begin{bmatrix} e \end{bmatrix} \right].$$

Our proof relies on the following lemmas.

**Lemma 4.2.1.** Let $(x, y)$ be such that $s_2(x, y) \succ_{\mathcal{K}_2} 0$. Then we have

$$0 \preceq Q^{xx} \preceq \nabla^2_{xx} v_2(x, y). \quad (4.2.1)$$

**Proof.** Let $h \in \mathbb{R}^{m_1}, h \neq 0$. We have

$$h^T Q^{xx} h = \sum_{i,j} Q_{ij} Q_{ij} h_i h_j$$

$$= (WH^{-1}W^T) \cdot \sum_{i,j} \left( \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, t_j \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, t_i \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} h_i h_j \right)$$

$$= (WH^{-1}W^T) \cdot \begin{bmatrix} s_2^{-1} \end{bmatrix} \left( \sum_{i,j} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, h_j t_j \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, h_i t_i \end{bmatrix} \right) \begin{bmatrix} s_2^{-1} \end{bmatrix}$$

$$= (WH^{-1}W^T) \cdot \begin{bmatrix} s_2^{-1} \end{bmatrix} \left( \sum_j \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, h_j t_j \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, h_i t_i \end{bmatrix} \right) \begin{bmatrix} s_2^{-1} \end{bmatrix}$$

$$= (WH^{-1}W^T) \cdot \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix}$$
\[
\begin{align*}
&= \left(\begin{bmatrix} s_2^{-1/2} \end{bmatrix} W H^{-1} W^T \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \right) \bullet \left(\begin{bmatrix} s_2^{-1/2} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \right)^2 \\
&= P \bullet \begin{bmatrix} \bar{b}^2 \end{bmatrix}.
\end{align*}
\]

Similarly we have

\[
\begin{align*}
\mathbf{h}^T R^{xx} \mathbf{h} &= (W H^{-1} W^T) \bullet \left( 2 \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} - \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \right) \\
&= \left(\begin{bmatrix} s_2^{-1/2} \end{bmatrix} W H^{-1} W^T \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \right) \\
&\quad \bullet \left(2 \left(\begin{bmatrix} s_2^{-1/2} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \right)^2 \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \begin{bmatrix} b \end{bmatrix} \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \right) \\
&= P \bullet \begin{bmatrix} \bar{b} \end{bmatrix},
\end{align*}
\]

and

\[
\begin{align*}
\mathbf{h}^T T^{xx} \mathbf{h} &= (W H^{-1} W^T) \bullet \begin{bmatrix} s_2 \end{bmatrix}^{-1} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} W H^{-1} W \begin{bmatrix} s_2^{-1} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1} \end{bmatrix} \\
&= P \bullet \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1/2} \end{bmatrix} P \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \begin{bmatrix} s_2, b \end{bmatrix} \begin{bmatrix} s_2^{-1/2} \end{bmatrix} \\
&= P \bullet \begin{bmatrix} \bar{b} \end{bmatrix} P \begin{bmatrix} \bar{b} \end{bmatrix}.
\end{align*}
\]

Using Proposition 4.2.1 and observing that \( P \succeq 0 \) and \( \begin{bmatrix} \bar{b}^2 \end{bmatrix} \succeq 0 \), we conclude that \( Q^{xx} \succeq 0 \).

Since \( P \) is a projection, we have that \( I - P \succeq 0 \) and therefore

\[
\begin{bmatrix} \bar{b} \end{bmatrix} P \begin{bmatrix} \bar{b} \end{bmatrix} \preceq \begin{bmatrix} \bar{b} \end{bmatrix}^2 = \frac{1}{2} \left( \begin{bmatrix} \bar{b}^2 \end{bmatrix} + \begin{bmatrix} \bar{b} \end{bmatrix} \right).
\]

This implies that

\[
P \bullet \begin{bmatrix} \bar{b} \end{bmatrix} P \begin{bmatrix} \bar{b} \end{bmatrix} \preceq \frac{1}{2} P \bullet \left( \begin{bmatrix} \bar{b}^2 \end{bmatrix} + \begin{bmatrix} \bar{b} \end{bmatrix} \right),
\]

which is exactly \( \mathbf{h}^T T^{xx} \mathbf{h} \preceq \frac{1}{2} \mathbf{h}^T (Q^{xx} + R^{xx}) \mathbf{h} \). Since \( \mathbf{h} \) is arbitrary, we have shown that \( T^{xx} \preceq \frac{1}{2} (Q^{xx} + R^{xx}) \), which together with \( Q^{xx} \succeq 0 \) establishes the result. \( \square \)
Lemma 4.2.2. For any $h \in \mathbb{R}^{m_1}$, and $(x, y)$ be such that $s_2(x, y) \succ_{\mathcal{K}_{x_2}} 0$. Then

$$
\| \bar{b} \| \leq \sqrt{2} r_2^{3/2} (h^\top Q^{xx} h)^{1/2}.
$$

(4.2.3)

Proof. By Theorem 1.3.1, there exist real numbers $\lambda_1, \lambda_2, \ldots, \lambda_{r_2}$ and a Jordan frame $c_1, c_2, \ldots, c_{r_2}$ such that $\bar{b} = \lambda_1 c_1 + \lambda_2 c_2 + \cdots + \lambda_{r_2} c_{r_2}$, and $\lambda_1, \lambda_2, \ldots, \lambda_{r_2}$ are the eigenvalues of $\bar{b}$. Without loss of generality (scaling $h$ as needed, and re-ordering indices), we may assume that $1 = |\lambda_1| \geq |\lambda_2| \geq \ldots \geq \lambda_{r_2}$. In view of Lemma 1.3.3, the matrix $[\bar{b}^2]$ has a full set of orthonormal eigenvectors $c_{ij}$ with corresponding eigenvalues $(1/2)(\lambda_i^2 + \lambda_j^2)$, for $1 \leq i \leq j \leq r_2$. It follows that

$$
\begin{align*}
\begin{aligned}
h^\top Q^{xx} h &= \frac{1}{2} P \left[ \sum_{i,j=1}^{r_2} (\lambda_i^2 + \lambda_j^2) c_{ij} c_{ij}^\top \right] \\
&= \frac{1}{2} \sum_{i,j=1}^{r_2} (\lambda_i^2 + \lambda_j^2) c_{ij}^\top P c_{ij}.
\end{aligned}
\end{align*}
$$

Recall that $P$ is a projection onto an $m_2$-dimensional space. So, we can write $P$ as

$$
P = \sum_{l=1}^{m_2} u_l u_l^\top,
$$

where $u_1, u_2, \ldots, u_{m_2}$ are the orthonormal eigenvectors of $P$ corresponding to the nonzero eigenvalues of $P$. Consider $u_k$ for some $k$, we have

$$
u_k = \sum_{i,j=1}^{r_2} \alpha_{ij} c_{ij},
$$

for some constants $\alpha_{ij}$, for $i, j = 1, 2, \ldots, r_2$, and

$$
1 = \| u_k \| = \left\| \sum_{i,j=1}^{r_2} \alpha_{ij} c_{ij} \right\| \leq \sum_{i,j=1}^{r_2} \| \alpha_{ij} c_{ij} \| = \sum_{i,j=1}^{r_2} | \alpha_{ij} |.
$$
This means that there exist $i_k$, $j_k$ such that

$$|\alpha_{i_k,j_k}| \geq \frac{1}{r_2^2}.$$ 

Thus

$$h^TQ^{xx}h = \frac{1}{2} \sum_{i,j} (\lambda_i^2 + \lambda_j^2) c_{ij}^T \left( \sum_{l=1}^{m_2} u_l u_l^T \right) c_{ij}$$

$$= \frac{1}{2} \sum_{i,j} (\lambda_i^2 + \lambda_j^2) \sum_{l=1}^{m_2} c_{ij}^T u_l u_l^T c_{ij}$$

$$= \frac{1}{2} \sum_{i,j} (\lambda_i^2 + \lambda_j^2) \| u_i^T c_{ij} \|^2$$

$$\geq \frac{1}{2} \sum_{i,j} (\lambda_i^2 + \lambda_j^2) |\alpha_{i_k,j_k}|^2$$

$$\geq \frac{1}{2} \sum_{i,j} (\lambda_i^2 + \lambda_j^2) \frac{1}{r_2^2}$$

$$\geq \frac{1}{2r_2^4} \sum_{j} (\lambda_i^2 + \lambda_j^2)$$

$$= \frac{1}{2r_2^4} \sum_{j} \| \tilde{b} \|^2$$

$$= \frac{1}{2r_2^3} \| \tilde{b} \|^2.$$ 

The result is established. $\Box$

We will next compute the third partial derivative of $v_2(x, y)$ with respect to $x$. To start, let $(x, y)$ be such that $s_2(x, y) \succ_{\text{aff}} 0$, and $h \in \mathbb{R}^{m_2}$. We have

$$\frac{\partial}{\partial x_i} h^T Q^{xx} h = W \left( \frac{\partial}{\partial x_i} H^{-1} \right) W^T \bullet \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right]$$

$$+ \left( WH^{-1} W^T \right) \bullet \frac{\partial}{\partial x_i} \left( \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right] \right)$$

$$= 2 WH^{-1} W^T \left[ s_2 \right]^{-1} \left[ s_2, t_i \right] \left[ s_2 \right]^{-1} WH^{-1} W^T \bullet \left( \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right] \right)$$

$$+ \left( WH^{-1} W^T \right) \bullet \frac{\partial}{\partial x_i} \left( \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right] \left[ s_2, b \right] \left[ s_2^{-1} \right] \right),$$
where

\[
\frac{\partial}{\partial x_i} \left( \left[ s_2^{-1} [s_2, b] [s_2^{-1} [s_2, b] [s_2^{-1} [s_2, b] \right) = -2 \left[ s_2^{-1} \right] \left( [s_2, t_i] [s_2^{-1} [s_2, b] \right] [s_2, b] \\
+ [s_2, b] [s_2^{-1} [s_2, t_i] [s_2^{-1} [s_2, b] \\
+ [s_2, b] [s_2^{-1} [s_2, [s_2^{-1} [s_2, t_i] \right] [s_2^{-1} [s_2, b] \\
+ [s_2^{-1} \left( [t_i, b] \right] [s_2^{-1} [s_2, b] \\
+ [s_2, b] [s_2^{-1} [t_i, b] \right] [s_2^{-1} \right].
\]

We conclude that the first directional derivative of \( h^\top Q^{xx} h \) with respect to \( x \), in the direction \( h \), is given by

\[
\nabla_x h^\top Q^{xx} h [h] = \sum_{i=1}^{m_1} h_i \frac{\partial}{\partial x_i} h^\top Q^{xx} h \\
= 2P \bullet [\bar{b}] P [\bar{b}^2] - 3P \bullet [\bar{b}^3] - P [\bar{b}^2, \bar{b}] .
\]

By following similar steps as above, we obtain

\[
\nabla_x h^\top R^{xx} h [h] = 2P \bullet [\bar{b}] P [\bar{b}] - 4P \bullet [\bar{b}^2, \bar{b}] ,
\]

\[
\nabla_x h^\top T^{xx} h [h] = 4P \bullet [\bar{b}] P [\bar{b}] P [\bar{b}] - 4P \bullet [\bar{b}] P [\bar{b}^2] - 2P \bullet [\bar{b}] P [\bar{b}] .
\]

By combining the previous results, we get

\[
\nabla^3_{xxx} v_2 (x, y) [h, h, h] = 12P \bullet [\bar{b}] P [\bar{b}^2] - 6P \bullet [\bar{b}^3] - 6P \bullet [\bar{b}^2, \bar{b}] \\
+ 6P \bullet [\bar{b}] P [\bar{b}] B - 8P \bullet [\bar{b}] P [\bar{b}] P [\bar{b}] .
\]

We need the following lemma which bounds \( \nabla^3_{xxx} v_2 (x, y) [h, h, h] \).

**Lemma 4.2.3.** For any \( h \in \mathbb{R}^{m_1} \) and \( (x, y) \) be such that \( s_2 (x, y) \geq K_{\gamma_2} 0 \). Then

\[
|\nabla^3_{xxx} v_2 (x, y) [h, h, h] | \leq \text{30} \| \bar{b} \| h^\top Q^{xx} h .
\]

90
**Proof.** Note that

$$\lfloor \bar{b} \rfloor \lfloor \bar{b} \rfloor = \frac{1}{2} (\lfloor \bar{b}^3 \rfloor + \lfloor \bar{b}^2, \bar{b} \rfloor).$$

Then (4.2.4) can be rewritten as

$$\nabla_{x, y, z}^3 v_2(x, y, h, h, h) = P \lfloor \bar{b} \rfloor P \cdot (12 \lfloor \bar{b}^2 \rfloor + 6 \lfloor \bar{b} \rfloor P \lfloor \bar{b} \rfloor) - 12 P \cdot \lfloor \bar{b}^2 \rfloor \lfloor \bar{b} \rfloor.$$

(4.2.6)

From (4.2.2) we have

$$12 \lfloor \bar{b}^2 \rfloor + 6 \lfloor \bar{b} \rfloor P \lfloor \bar{b} \rfloor \succeq 8 \lfloor \bar{b}^2 \rfloor + 2 \lfloor \bar{b} \rfloor.$$

By observing that that $-\lfloor \bar{b}^2 \rfloor \preceq \lfloor \bar{b} \rfloor \preceq \lfloor \bar{b}^2 \rfloor$, we obtain

$$6 \lfloor \bar{b}^2 \rfloor \preceq 12 \lfloor \bar{b}^2 \rfloor + 6 \lfloor \bar{b} \rfloor P \lfloor \bar{b} \rfloor \preceq 18 \lfloor \bar{b}^2 \rfloor.$$

(4.2.7)

Let $\lambda_1, \lambda_2, \ldots, \lambda_{r_2}$ be the eigenvalues of $\bar{B}$. Then, for $i, j = 1, 2, \ldots, r_2$, the eigenvalues of $\lfloor \bar{b} \rfloor$ are of the form $(1/2)(\lambda_i + \lambda_j)$ (see Lemma 1.3.3). We then have

$$-\|\bar{b}\| I \preceq \lfloor \bar{b} \rfloor \preceq \|\bar{b}\| I$$

and hence

$$-\|\bar{b}\| P \preceq P \lfloor \bar{b} \rfloor P \preceq \|\bar{b}\| P.$$  

(4.2.8)

Using (4.2.7), (4.2.8), and the fact that $\lfloor \bar{b}^2 \rfloor \succeq 0$, we get

$$\|P \lfloor \bar{b} \rfloor P \cdot (12 \lfloor \bar{b}^2 \rfloor + 6 \lfloor \bar{b} \rfloor P \lfloor \bar{b} \rfloor)\| \leq 18 \|\bar{b}\| P \cdot \lfloor \bar{b}^2 \rfloor.$$

(4.2.9)

In addition, since the elements $\bar{b}^2$ and $\bar{b}$ have the same eigenvectors, the matrices $\lfloor \bar{b}^2 \rfloor$
and $\bar{b}$ also have the same eigenvectors. This implies that

$$-\|\bar{b}\| \bar{b}^2 \preceq \bar{b}^2 \preceq \|\bar{b}\| \bar{b}^2.$$ 

Thus we have that

$$\left\| P \cdot \bar{b}^2 \right\| \leq \|\bar{b}\| P \cdot \bar{b}^2. \quad (4.2.10)$$

The result follows from (4.2.6), (4.2.9) and (4.2.10). \(\Box\)

We can now state the proof of Theorem 4.2.1.

**Theorem 4.2.1.** For any fixed $\mu > 0$, $\rho_k(\mu, \cdot)$ is $\mu$-self-concordant on $F_0$, for $k = 1, 2, \ldots, K$.

**Proof.** By combining the results of (4.2.1), (4.2.3) and (4.2.5), we get

$$\left| \nabla^3_{xxx} v_2(x, \bar{y}) [h, h, h] \right| \leq 30\sqrt{2}\eta_2^{3/2}(h^T \nabla^2_{xx} v_2(x, \bar{y}) h)^{3/2},$$

which combined with (4.1.14) implies that

$$\left| \nabla^3_{xxx} \rho_k(y) [h, h, h] \right| \leq 30\sqrt{2}\mu c_2 r_2^{3/2}(\nabla^2_{xx} v_2(x, \bar{y}) [h, h])^{3/2}$$

$$= 2\mu^{-1/2}(c_2 \mu \nabla^2_{xx} v_2(x, y) [h, h])^{3/2}$$

$$= 2\mu^{-1/2}(\nabla^2_{xx} \rho_k(x) [h, h])^{3/2}.$$

The theorem is established. \(\Box\)

**Corollary 4.2.1.** For any fixed $\mu > 0$, $\eta(\mu, \cdot)$ is a $\mu$-self-concordant function on $F_0$.

**Proof.** It is easy to verify that $\mu c_1 v_1$ is $\mu$-self-concordant on $F_1$. The corollary follows from [30, Proposition 2.1.1]. \(\Box\)
4.2.2 Parameters of the self-concordant family

In this subsection, we show that the family of functions \( \{ \eta(\mu, \cdot) : \mu > 0 \} \) is a strongly self-concordant family with appropriate parameters (see definition 3.2.2).

The proof of self-concordancy of the family \( \{ \eta(\mu, \cdot) : \mu > 0 \} \) relies on the following two lemmas.

**Lemma 4.2.4.** For any \( \mu > 0 \) and \( x \in F_0 \), the following inequality holds:

\[
|\{ \nabla_{xx}^2 \eta(\mu, x) \}'(h, h) | \leq \frac{1}{\mu} \nabla_{xx}^2 \eta(\mu, x)(h, h), \quad \forall h \in \mathbb{R}^m.
\]

**Proof.** Differentiating \( \nabla_{xx}^2 \eta(\mu, x) \) in (4.1.15) with respect to \( \mu \), we obtain

\[
\{ \nabla_{xx}^2 \eta(\mu, x) \}' = \nabla_{xx}^2 v_1(x) + \sum_{k=1}^{K} \{ \nabla_{xx}^2 v_2^{(k)}(x, \bar{y}^{(k)}) + \mu \nabla_{xx}^3 v_2^{(k)}(x, \bar{y}^{(k)}) \circ (\bar{y}^{(k)})' \}
\]

\[
= \nabla_{xx}^2 v_1(x) + \sum_{k=1}^{K} \nabla_{xx}^2 v_2^{(k)}(x, \bar{y}^{(k)}) = \frac{1}{\mu} \nabla_{xx}^2 \eta(\mu, x).
\]

The result immediately follows by observing that \( \nabla_{xx}^2 \eta(\mu, x) \succeq 0 \), and therefore, for any \( h \in \mathbb{R}^n \), we have that \( \frac{1}{\mu} \nabla_{xx}^2 \eta(\mu, x)(h, h) \geq 0 \). \( \square \)

For fixed \( (x, \bar{y}) \) with \( s_2(x, \bar{y}) \succeq_{K_2} 0 \), let

\[
\bar{t}_i = \left[ s_2^{-1/2} \right] \left[ s_2, t_i \right] \left[ s_2^{-1/2} \right] e_2 \quad \text{and} \quad \bar{w}_j = \left[ s_2^{-1/2} \right] \left[ s_2, w_j \right] \left[ s_2^{-1/2} \right] e_2,
\]

for \( i = 1, 2, \ldots, m_1 \) and \( j = 1, 2, \ldots, m_2 \). We can apply a Gram-Schmidt procedure to \( \{ [\bar{w}_i] \} \) and obtain \( \{ [\bar{u}_i] \} \) with \( \| [\bar{u}_i] \| = 1 \) for all \( i \) and \( \bar{u}_i \cdot \bar{u}_j = 0, i \neq j \). Then the linear span of \( \{ [\bar{u}_i], i = 1, 2, \ldots, m_2 \} \) is equal to the span of \( \{ [\bar{w}_i], i = 1, 2, \ldots, m_2 \} \).

Let \( \bar{U} = [\bar{u}_1; \bar{u}_2; \ldots; \bar{u}_{m_2}] \in \mathbb{R}^{n_2 \times m_2} \) and \( \hat{u} = \sum_{k=1}^{m_2} \bar{u}_i \). It follows that \( P = UU^\top \). We
then have

\[
\frac{\partial v_2(x, \tilde{y})}{\partial x_i} = -P \cdot [\tilde{t}_i]
\]

\[
= -UU^T \cdot [\tilde{t}_i]
\]

\[
= -\text{trace}(U [\tilde{t}_i] U)
\]

\[
= - \sum_{k=1}^{m_2} u_k \cdot (\tilde{t}_i \circ u_k)
\]

\[
= - \sum_{k=1}^{m_2} \tilde{t}_i \cdot \tilde{u}_k^2
\]

\[
= -\tilde{t}_i \cdot \sum_{k=1}^{m_2} \tilde{u}_k^2
\]

\[
= -\tilde{t}_i \cdot \hat{u},
\]

and

\[
Q_{i,j}^{xx} = P \cdot [\tilde{t}_i] [\tilde{t}_j]
\]

\[
= \text{trace}(U [\tilde{t}_i] [\tilde{t}_j] U)
\]

\[
= \sum_{k=1}^{m_2} (\tilde{u}_k \cdot (\tilde{t}_i \circ (\tilde{t}_j \circ \tilde{u}_k)))
\]

\[
= \sum_{k=1}^{m_2} \tilde{t}_i \cdot (\tilde{u}_k^2 \circ \tilde{t}_j)
\]

\[
= \tilde{t}_i \cdot \left( \sum_{k=1}^{m_2} \tilde{u}_k^2 \circ \tilde{t}_j \right)
\]

\[
= \tilde{t}_i \cdot (\hat{u} \circ \tilde{t}_j).
\]

Lemma 4.2.5. Let \((x, \tilde{y})\) be such that \(s_2(x, \tilde{y}) \succ K_j^2 0\). Then

\[
\nabla_x v_2(x, \tilde{y})^T \{\nabla_{xx}^2 v_2(x, \tilde{y})\}^{-1} \nabla_x v_2(x, \tilde{y}) \leq m_2.
\]

Proof. Let \(\tilde{T} = [\tilde{t}_1; \tilde{t}_2; \ldots; \tilde{t}_{m_1}] \in \mathbb{R}^{n_2^2 \times m_1}\). From (4.2.12) we have that

\[
Q_{i,j}^{xx} = \tilde{t}_i \cdot (\hat{u} \circ \tilde{t}_j) = \tilde{t}_i \cdot [\hat{u}] \tilde{t}_j,
\]

94
and hence

\[ Q^{xx} = \bar{T}^T \begin{bmatrix} \hat{u} \\ \bar{T} \end{bmatrix} \bar{T}. \]

From (4.2.11), we also have

\[ \nabla_x v_2(x, \bar{y})^T = -\bar{T}^T \hat{u}. \]

Thus

\[
\nabla_x v_2(x, \bar{y})^T (Q^{xx})^{-1} \nabla_x v_2(x, \bar{y}) = \hat{u} \cdot \bar{T}(\bar{T}^T \begin{bmatrix} \hat{u} \\ \bar{T} \end{bmatrix} \bar{T})^{-1} \bar{T}^T \begin{bmatrix} \hat{u} \\ \bar{T} \end{bmatrix} \bar{T}
\]

\[
\leq \hat{u}^1/2 \cdot \hat{u}^{1/2}
\]

\[
= \text{trace}(\hat{u})
\]

\[
= m_2,
\]

where the last equality follows from the fact that \( \hat{u} = \sum_{k=1}^{m_2} u_k \), and that \( \text{trace}(u_k^2) = u_k \cdot u_k = 1 \) for each \( k \). In addition, \( Q^{xx} \preceq \nabla^2_{xx} v_2(x, \bar{y}) \) implies \( \{\nabla^2_{xx} v_2(x, \bar{y})\}^{-1} \succeq (Q^{xx})^{-1} \).

This completes the proof. \( \square \)

**Lemma 4.2.6.** For any \( \mu > 0 \) and \( x \in F_0 \), we have

\[
|\nabla_x \eta'(\mu, x)^T h| \leq \sqrt{\frac{(m_1 c_1 + m_2 c_2)(1 + K)}{\mu} \nabla^2_{xx} \eta(\mu, x)[h, h]}, \quad \forall \ h \in \mathbb{R}^{m_1}.
\]

**Proof.** Differentiating \( \nabla_x \eta(\mu, x) \) in (4.1.15) with respect to \( \mu \), we obtain

\[
\nabla_x \eta'(\mu, x) = c_1 \nabla_x v_1(x) + \sum_{k=1}^{K} c_1 \{\nabla_x v_2^{(k)}(x, \bar{y}^{(k)}) + \mu \nabla^2_{xy} v_2^{(k)}(x, \bar{y}^{(k)}) \cdot (\bar{y}^{(k)})'\}
\]

\[
= c_1 \nabla_x v_1(x) + \sum_{k=1}^{K} c_2 \nabla_x v_2^{(k)}(x, \bar{y}^{(k)}).
\]

95
In Lemma 4.2.13, we have shown that
\[ \nabla_{x} v_{2}^{(k)}(x, \bar{y}^{(k)})^\intercal \{ \nabla_{xx}^2 v_{2}^{(k)}(x, \bar{y}^{(k)}) \}^{-1} \nabla_{x} v_{2}^{(k)}(x, \bar{y}^{(k)}) \leq m_2, \]
which is equivalent to
\[ |\nabla_{x} v_{2}^{(k)}(x, \bar{y}^{(k)})[h]| \leq \sqrt{m_2 \nabla_{xx}^2 v_{2}^{(k)}(x, \bar{y}^{(k)})[h, h]}, \quad \forall h \in \mathbb{R}^{m_2}. \quad (4.2.14) \]
Similarly, we can show that
\[ \nabla_{x} v_{1}(x)^\intercal \{ \nabla_{xx}^2 v_{1}(x) \}^{-1} \nabla_{x} v_{1}(x) \leq m_1, \]
which is equivalent to
\[ |\nabla_{x} v_{1}(x)[h]| \leq \sqrt{m_1 \nabla_{xx}^2 v_{1}(x)[h, h]}, \quad \forall h \in \mathbb{R}^{m_1}. \quad (4.2.15) \]
Then, using (4.2.14) and (4.2.15), we have that for all \( h \in \mathbb{R}^{m_1} \)
\[
|\nabla_{x} \eta'(\mu, x)[h]| = \left| c_1 \nabla_{x} v_{1}(x) + \sum_{k=1}^{K} c_2 \nabla_{x} v_{2}^{(k)}(x, \bar{y}^{(k)}) \right| [h] \\
\leq |c_1 \nabla_{x} v_{1}(x)[h]| + \sum_{k=1}^{K} |c_2 \nabla_{x} v_{2}^{(k)}(x, \bar{y}^{(k)})[h]| \\
\leq \sqrt{m_1 c_1^2 \nabla_{xx}^2 v_{1}(x)[h, h]} + \sum_{k=1}^{K} \sqrt{m_2 c_2^2 \nabla_{xx}^2 v_{2}^{(k)}(x, \bar{y}^{(k)})[h, h]} \\
\leq \sqrt{(m_1 c_1) c_1 \nabla_{xx}^2 v_{1}(x)[h, h]} + \sum_{k=1}^{K} \sqrt{(m_2 c_2) c_2 \nabla_{xx}^2 v_{2}^{(k)}(x, \bar{y}^{(k)})[h, h]} \\
\leq \sqrt{(m_1 c_1 + m_2 c_2)(1 + K)} \left( c_1 \nabla_{xx}^2 v_{1}(x)[h, h] + \sum_{k=1}^{K} c_2 \nabla_{xx}^2 v_{2}^{(k)}(x, \bar{y}^{(k)})[h, h] \right) \\
= \sqrt{(m_1 c_1 + m_2 c_2)(1 + K)} \mu \nabla_{xx}^2 \eta(\mu, x)[h, h].
\]
The result is established. □

**Theorem 4.2.2.** The family \( \{ \eta(\mu, \cdot) : \mu > 0 \} \) is a strongly self-concordant family with the following parameters

\[
\alpha_1(\mu) = \mu, \quad \alpha_2(\mu) = \alpha_3(\mu) = 1, \quad \alpha_4(\mu) = \frac{\sqrt{(1 + K)(m_1c_1 + m_2c_2)}}{\mu}, \quad \alpha_5(\mu) = \frac{1}{\mu}.
\]

**Proof.** It is direct to see that condition (i) of Definition 3.2.2 holds. Corollary 4.2.1 shows that condition (ii) is satisfied and Lemmas 4.2.4 and 4.2.6 show that condition (iii) is satisfied. □

### 4.3 A class of volumetric barrier algorithms for solving SSPs

In §5 we have established that the parametric functions \( \eta(\mu, \cdot) \) is a strongly self-concordant family. Therefore, it becomes straightforward to develop primal path following interior point algorithms for solving SSP (4.1.3, 4.1.4). In this section we introduce a class of volumetric barrier algorithms for solving this problem. This class is stated formally in Algorithm 2.

Our algorithm is initialized with a starting point \( x^0 \in F_0 \) and a starting value \( \mu^0 > 0 \) for the barrier parameter \( \mu \), and is indexed by a parameter \( \gamma \in (0, 1) \). We use \( \delta \) as a measure of the proximity of the current point \( x \) to the central path, and \( \beta \) as a threshold for that measure. If the current \( x \) is too far away from the central path in the sense that \( \delta > \beta \), Newton’s method is applied to find a point close to the central path. Then the value of \( \mu \) is reduced by a factor \( \gamma \) and the whole process is repeated until the value of \( \mu \) is within the tolerance \( \epsilon \). By tracing the central path as \( \mu \) approaches zero, a strictly
Algorithm 2 Volumetric Barrier Algorithm for Solving SSP (4.1.3,4.1.4)

Require: $\epsilon > 0$, $\gamma \in (0, 1)$, $\theta > 0$, $\beta > 0$, $x^0 \in F_0$ and $\mu^0 > 0$.

$x := x^0$, $\mu := \mu^0$

while $\mu \geq \epsilon$

for $k = 1, 2, \ldots, K$

solve (4.1.6) to obtain $\bar{y}^{(k)}$

end for

compute $\Delta x := -\{\nabla^2_{xx} \eta(\mu, x)\}^{-1}\nabla_x \eta(\mu, x)$ using (4.1.15)

compute $\delta(\mu, x) := \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \eta(\mu, x) \Delta x}$ using (4.1.15)

while $\delta > \beta$

$x := x + \theta \Delta x$

for $k = 1, 2, \ldots, K$

solve (4.1.6) to obtain $\bar{y}^{(k)}$

end for

compute $\Delta x := -\{\nabla^2_{xx} \eta(\mu, x)\}^{-1}\nabla_x \eta(\mu, x)$ using (4.1.15)

compute $\delta(\mu, x) := \sqrt{-\frac{1}{\mu} \Delta x^T \nabla^2_{xx} \eta(\mu, x) \Delta x}$ using (4.1.15)

end while

$\mu := \gamma \mu$

end while

feasible $\epsilon$-solution to (4.1.6) will be generated.

4.4 Complexity analysis

Theorems 4.4.1 and 4.4.2 present the complexity analysis for two variants of algorithms: short-step algorithms and long-step algorithms, which are controlled by the manner of selection $\gamma$ that we have made in Algorithm 2.

In the short-step version of the algorithm, we decrease the barrier parameter in each iteration by a factor $\gamma := 1 - \sigma / \sqrt{(1 + K)(m_1c_1 + m_2c_2)}$, $\sigma < 0.1$. The $k^{th}$ iteration of the short-step algorithms is performed as follows: at the beginning of the iteration, we have $\mu^{(k-1)}$ and $x^{(k-1)}$ on hand and $x^{(k-1)}$ is close to the center path, i.e., $\delta(\mu^{(k-1)}, x^{(k-1)}) \leq \beta$. After the barrier parameter $\mu$ is reduced from $\mu^{(k-1)}$ to $\mu^k := \gamma \mu^{(k-1)}$, we have that $\delta(\mu^k, x^{(k-1)}) \leq 2\beta$. Then a full Newton step with size $\theta = 1$ is taken to produce a new
point \( x^k \) with \( \delta(\mu^k, x^k) \leq \beta \). We now show that, in this class of algorithms, only one Newton step is sufficient for recentering after updating the parameter \( \mu \). We have the following theorem.

**Theorem 4.4.1.** Consider Algorithm 2 and let \( \mu^0 \) be the initial barrier parameter, \( \epsilon > 0 \) the stopping criterion, and \( \beta = (2 - \sqrt{3})/2 \). If the starting point \( x^0 \) is sufficiently close to the central path, i.e., \( \delta(\mu^0, x^0) \leq \beta \), then the short-step algorithm reduces the barrier parameter \( \mu \) at a linear rate and terminates with at most \( O(\sqrt{(1+K)(m_1c_1 + m_2c_2}) \ln(\mu^0/\epsilon)) \) iterations.

In the long-step version of the algorithm, the barrier parameter is decreased by an arbitrary constant factor \( \gamma \in (0, 1) \). It has a potential for larger decrease on the objective function value, however, several damped Newton steps might be needed for restoring the proximity to the central path. The \( k^{th} \) iteration of the long-step algorithms is performed as follows: at the beginning of the iteration we have a point \( x^{(k-1)} \), which is sufficiently close to \( x(\mu^{(k-1)}) \), where \( x(\mu^{(k-1)}) \) is the solution to (4.1.5) for \( \mu := \mu^{(k-1)} \). The barrier parameter is reduced from \( \mu^{(k-1)} \) to \( \mu^k := \gamma \mu^{(k-1)} \), where \( \gamma \in (0, 1) \), and then searching is started to find a point \( x^k \) that is sufficiently close to \( x(\mu^k) \). The long-step algorithm generates a finite sequence consisting of \( N \) points in \( F_0 \), and we finally take \( x^k \) to be equal to the last point of this sequence. We want to determine an upper bound on \( N \), the number of Newton iterations that are needed to find the point \( x^k \). We have the following theorem.

**Theorem 4.4.2.** Consider Algorithm 2 and let \( \mu^0 \) be the initial barrier parameter, \( \epsilon > 0 \) the stopping criterion, and \( \beta = 1/6 \). If the starting point \( x^0 \) is sufficiently close to the central path, i.e., \( \delta(\mu^0, x^0) \leq \beta \), then the long-step algorithm reduces the barrier parameter \( \mu \) at a linear rate and terminates with at most \( O((1+K)(m_1c_1 + m_2c_2) \ln(\mu^0/\epsilon)) \) iterations.

The proofs of the Theorems 4.4.1 and 4.4.1 are similar to the proofs of Theorems 3.4.1
and 3.4.2 and are given in the subsections below. For convenience, we restate Proposition
3.4.1(i) and Lemma 3.4.1.

**Proposition 4.4.1.** For any $\mu > 0$ and $x \in F_0$, we denote $\Delta x := -\{\nabla^2_{xx} \eta(\mu, x)\}^{-1} \nabla_x \eta(\mu, x)$ and $\delta := \sqrt{\frac{1}{\mu} \nabla^2_{xx} \eta(\mu, x)[\Delta x, \Delta x]}$. Then for $\delta < 1$, $\tau \in [0, 1]$ and any $h \in \mathbb{R}^{m_2}$ we have
\[
\nabla^2_{xx} \eta(\mu, x + \tau \Delta x)[h, h] \leq (1 - \tau \delta)^2 \nabla^2_{xx} \eta(\mu, x)[h, h].
\]

**Lemma 4.4.1.** For any $\mu > 0$ and $x \in F_0$, let $\Delta x$ be the Newton direction defined by $\Delta x := -\{\nabla^2_{xx} \eta(\mu, x)\}^{-1} \nabla_x \eta(\mu, x)$; $\delta := \delta(\mu, x) = \sqrt{\frac{1}{\mu} \nabla^2_{xx} \eta(\mu, x)[\Delta x, \Delta x]}$, $x^+ = x + \Delta x$, $\Delta x^+$ be the Newton direction calculated at $x^+$, and $\delta(\mu, x^+) := \sqrt{\frac{1}{\mu} \nabla^2_{xx} \eta(\mu, x^+)[\Delta x^+, \Delta x^+]}$. Then the following relations hold:

(i) If $\delta < 2 - \sqrt{3}$, then $\delta(\mu, x^+) \leq \frac{\delta}{1 - \delta} \leq \frac{\delta}{2}$.

(ii) If $\delta \geq 2 - \sqrt{3}$, then $\eta(\mu, x) - \eta(\mu, x + \bar{\theta} \Delta x) \geq \mu(\delta - \ln(1 + \delta))$, where $\bar{\theta} = (1 + \delta)^{-1}$.

### 4.4.1 Complexity for short-step algorithm

For the purpose of proving Theorems 4.4.1, we present the following proposition which is a restatement of [30, Theorem 3.1.1].

**Proposition 4.4.2.** Let $\chi_\kappa(\eta; \mu, \mu^+) := \left(1 + \frac{\sqrt{(1+K)(m_1c_1+m_2c_2)\kappa}}{\kappa}\right) \ln \gamma^{-1}$. Assume that $\delta(\mu, x) < \kappa$ and $\mu^+ := \gamma \mu$ satisfies
\[
\chi_\kappa(\eta; \mu, \mu^+) \leq 1 - \frac{\delta(\mu, x)}{\kappa}.
\]
Then $\delta(\mu^+, x) < \kappa$.

**Lemma 4.4.2.** Let $\mu^+ = \gamma \mu$, where $\gamma = 1 - \sigma/\sqrt{(1+K)(m_1c_1+m_2c_2)}$ and $\sigma \leq 0.1$, and let $\beta = (2 - \sqrt{3})/2$. If $\delta(\mu, x) \leq \beta$, then $\delta(\mu^+, x) \leq 2\beta$. 

100
Proof. Let $\kappa := 2\beta = 2 - \sqrt{3}$. Since $\delta(\mu, \mathbf{x}) \leq \kappa/2$, one can verify that for $\sigma \leq 0.1$, $\mu^+$ satisfies

$$\chi(\eta; \mu, \mu^+) \leq \frac{1}{2} \leq 1 - \frac{\delta(\mu, \mathbf{x})}{\kappa}.$$ 

By Proposition 4.4.2, we have $\delta(\mu^+, \mathbf{x}) \leq \kappa$. \qed

By Lemmas 4.4.1(i) and 4.4.2, we conclude that we can reduce the parameter $\mu$ by the factor $\gamma := 1 - \sigma/\sqrt{(1+K)(m_1c_1 + m_2c_2)}$, $\sigma < 0.1$, at each iteration, and that only one Newton step is sufficient to restore proximity to the central path. Hence, Theorem 4.4.1 follows.

### 4.4.2 Complexity for long-step algorithm

For $\mathbf{x} \in \mathcal{F}_0$ and $\mu > 0$, we define the function $\phi(\mu, \mathbf{x}) := \eta(\mu, \mathbf{x}(\mu)) - \eta(\mu, \mathbf{x})$, which represents the difference between the objective value $\eta(\mu^k, \mathbf{x}^{(k)})$ at the end of $k^{th}$ iteration and the minimum objective value $\eta(\mu^k, \mathbf{x}^{(k-1)})$ at the beginning of $k^{th}$ iteration. Then the task is to find an upper bound on $\phi(\mu, \mathbf{x})$. To do so, we first give upper bounds on $\phi(\mu, \mathbf{x})$ and $\phi'(\mu, \mathbf{x})$ respectively. We have the following lemma.

**Lemma 4.4.3.** Let $\mu > 0$ and $\mathbf{x} \in \mathcal{F}_0$, we denote $\tilde{\Delta} \mathbf{x} := \mathbf{x} - \mathbf{x}(\mu)$ and define

$$\tilde{\delta} := \tilde{\delta}(\mu, \mathbf{x}) = \sqrt{\frac{1}{\mu} \nabla^2_{\mathbf{x}\mathbf{x}} \eta(\mu, \mathbf{x})[\tilde{\Delta} \mathbf{x}, \tilde{\Delta} \mathbf{x}].$$

For any $\mu > 0$ and $\mathbf{x} \in \mathcal{F}_0$, if $\tilde{\delta} < 1$, then the following inequalities hold:

$$\phi(\mu, \mathbf{x}) \leq \mu \left( \frac{\tilde{\delta}}{1 - \tilde{\delta}} + \ln(1 - \tilde{\delta}) \right), \quad (4.4.1)$$

$$|\phi'(\mu, \mathbf{x})| \leq -\sqrt{(1+K)(m_1c_1 + m_2c_2)} \ln(1 - \tilde{\delta}). \quad (4.4.2)$$
Proof.

\[ \phi(\mu, x) := \eta(\mu, x) - \eta(\mu, x(\mu)) := \int_0^1 \nabla_x \eta(\mu, x + (1 - \tau)\Delta x)(\Delta x) d\tau. \]

Since \( x(\mu) \) is the optimal solution, we have

\[ \nabla_x \eta(\mu, x(\mu)) = 0. \]  \hfill (4.4.3)

Hence, with the aid of Proposition 4.4.1, we get

\[
\phi(\mu, x) = \int_0^1 \int_0^1 \nabla_{xx} \eta(\mu, x(\mu) + (1 - \alpha)\Delta x)[\Delta x, \Delta x] d\alpha d\tau \\
\leq \int_0^1 \int_0^1 \frac{\nabla_{xx} \eta(\mu, x)[\Delta x, \Delta x]}{(1 - \hat{\delta} + \alpha \hat{\delta})^2} d\alpha d\tau \\
= \int_0^1 \int_0^1 \frac{\mu \hat{\delta}^2}{(1 - \hat{\delta} + \alpha \hat{\delta})^2} d\alpha d\tau \\
= \mu \left( \frac{\hat{\delta}}{1 - \hat{\delta} + \ln(1 - \hat{\delta})} \right),
\]

which establishes (4.4.1).

Now, for any \( \mu > 0 \), by applying the chain rule, using (4.4.3), and applying the Mean-Value Theorem, we get

\[
\phi'(\mu, x) = \eta'(\mu, x) - \eta'(\mu, x(\mu)) - \nabla_x \eta(\mu, x(\mu))^T x'(\mu) \\
= \eta'(\mu, x) - \eta'(\mu, x(\mu)) \\
= \nabla_x \eta(\mu, x(\mu) + \omega \Delta x)^T \Delta x, \]  \hfill (4.4.4)

for some \( \omega \in (0, 1) \). Hence

\[
|\phi'(\mu, x)| = \left| \int_0^1 \{\nabla_x \eta(\mu, x(\mu) + \tau \Delta x)\}'[\Delta x] d\tau \right|
\]

102
\[
\int_0^1 \sqrt{\nabla^2_{xx} \eta(\mu, x(\mu) + \tau \Delta x)} [\Delta x, \Delta x] \sqrt{\{\nabla_x \eta(\mu, x(\mu) + \tau \Delta x)\}' \cdot (\{\nabla^2_{xx} \eta(\mu, x(\mu) + \tau \Delta x)\}^{-1} \{\nabla_x \eta(\mu, x(\mu) + \tau \Delta x)\})'} \, d\tau.
\]

In view of Lemma 4.2.6 we have the following estimation

\[
-\{\nabla_x \eta(\mu, x)\}' \cdot (\{\nabla^2_{xx} \eta(\mu, x)\}^{-1} \{\nabla_x \eta(\mu, x)\})' \leq \frac{(1 + K)(m_1 c_1 + m_2 c_2)}{\mu}.
\] (4.4.5)

Then, by using (4.4.5), Proposition 4.4.1, and the observation \(x(\mu) + \tau \Delta x = x - (1 - \tau) \Delta x\), we obtain

\[
|\phi'(\mu, x)| \leq \int_0^1 \sqrt{\nabla^2_{xx} \eta(\mu, x)[\Delta x, \Delta x]} \sqrt{\frac{(1 + K)(m_1 c_1 + m_2 c_2)}{\mu}} d\tau = \int_0^1 \frac{\tilde{\delta} \sqrt{\mu}}{1 - \tilde{\delta} + \tau \tilde{\delta}} \sqrt{\frac{(1 + K)(m_1 c_1 + m_2 c_2)}{\mu}} d\tau = \sqrt{(1 + K)(m_1 c_1 + m_2 c_2)} \int_0^1 \frac{\tilde{\delta}}{1 - \tilde{\delta} + \tau \tilde{\delta}} d\tau = -\sqrt{(1 + K)(m_1 c_1 + m_2 c_2)} \ln(1 - \tilde{\delta}),
\]

which establishes (4.4.2). \(\square\)

**Lemma 4.4.4.** Let \(\mu > 0\) and \(x \in F_0\) be such that \(\tilde{\delta} < 1\), where \(\tilde{\delta}\) is as defined in Lemma 4.4.3. Let \(\mu^+ := \gamma \mu\) with \(\gamma \in (0, 1)\). Then

\[
\eta(\mu^+, x) - \eta(\mu^+, x(\mu^+)) \leq O(1)[(1 + K)(m_1 c_1 + m_2 c_2)]\mu^+.
\]

**Proof.** Differentiating (4.4.4) with respect to \(\mu\), we get

\[
\phi''(\mu, x) = \eta''(\mu, x) - \eta''(\mu, x(\mu)) - \{\nabla_x \eta(\mu, x(\mu))\}' x'(\mu). \tag{4.4.6}
\]

We will work on the right-hand side of (4.4.6) by bounding the second and the last terms. Observe that \(\eta''(\mu, x(\mu)) = \sum_{k=1}^K \rho^{(k)''}(\mu, x)\). Differentiating \(\rho^{(k)}(\mu, x)\) with re-
spect to $\mu$ we obtain $\rho^{(k)'}(\mu, x) = c_2 v_2(x, \tilde{y})$. Now, differentiating $\rho^{(k)'}(\mu, x)$ with respect to $\mu$ we obtain $\rho^{(k)''}(\mu, x) = c_2 \nabla_{\tilde{y}} v_2(x, \tilde{y})^T \tilde{y}'$. Then, differentiating (4.1.13) yields

$$\tilde{y}' = -\frac{1}{\mu} \{\nabla_{yy}^2 v_2(x, \tilde{y})\}^{-1} \nabla_{\tilde{y}} v_2(x, \tilde{y}).$$

Therefore we have

$$\rho^{(k)''}(\mu, x) = -\frac{1}{\mu} c_2 \nabla_{\tilde{y}} v_2(x, \tilde{y})^T \{\nabla_{yy}^2 v_2(x, \tilde{y})\}^{-1} \nabla_{\tilde{y}} v_2(x, \tilde{y}).$$

It can be shown (see also [7, Theorem 4.4]) that $-\rho^{(k)''}(\mu, y) \leq \frac{1}{\mu} c_2 m_2$. Thus

$$-\eta''(\mu, x(\mu)) = -\sum_{k=1}^{K} \rho^{(k)''}(\mu, x) \leq \frac{m_2 c_2 K}{\mu r_2}. \tag{4.4.7}$$

By differentiating (4.4.3) with respect to $\mu$, we get

$$\{\nabla_x \eta(\mu, x(\mu))\}' + \nabla_{xx}^2 \eta(\mu, x(\mu)) x'(\mu) = 0,$$

or equivalently,

$$x'(\mu) = -\{\nabla_{xx}^2 \eta(\mu, x(\mu))\}^{-1} \nabla_x \eta'(\mu, x(\mu)).$$

Hence, by using (4.4.5), we have

$$-\{\nabla_x \eta(\mu, x(\mu))\}'^T x'(\mu) = \{\nabla_x \eta(\mu, x(\mu))\}'^T \{\nabla_{xx}^2 \eta(\mu, x(\mu))\}^{-1} \nabla_x \{\eta(\mu, x(\mu))\}' \leq \frac{(1 + K)(m_1 c_1 + m_2 c_2)}{\mu}. \tag{4.4.8}$$

Observe that $\eta(\mu, x)$ is concave in $\mu$, it follows that $\eta''(\mu, x) \leq 0$. Combining this with (4.4.7) and (4.4.8), we obtain

104
\[ \phi''(\mu, x(\mu)) \leq \frac{m(1 + K)(m_1 c_1 + 2m_2 c_2)}{\mu}. \]  

(4.4.9)

Applying the Mean-Value Theorem and using Lemma 4.4.3 and (4.4.9) to get

\[
\phi(\mu^+, x) = \phi(\mu, x) + \phi'(\mu, x)(\mu^+ - \mu) + \int_{\mu}^{\mu^+} \int_\tau^{\mu} \phi''(v, x) \, dv \, d\tau
\]

\[
\leq \mu \left( \frac{\tilde{\delta}}{1 - \tilde{\delta}} + \ln(1 - \tilde{\delta}) \right) - \sqrt{(1 + K)(m_1 c_1 + m_2 c_2)} \ln(1 - \tilde{\delta}) (\mu^+ - \mu)
\]

\[
+ m(1 + K)(m_1 c_1 + 2m_2 c_2) \int_{\mu}^{\mu^+} \int_\tau^{\mu} v^{-1} \, dv \, d\tau
\]

\[
= \mu \left( \frac{\tilde{\delta}}{1 - \tilde{\delta}} + \ln(1 - \tilde{\delta}) \right) - \sqrt{(1 + K)(m_1 c_1 + m_2 c_2)} \ln(1 - \tilde{\delta}) (\mu^+ - \mu)
\]

\[
+ m(1 + K)(m_1 c_1 + 2m_2 c_2) (\mu - \mu^+) \ln \tilde{\delta} \mu
\]

\[
\leq \mu \left( \frac{\delta}{1 - \delta} + \ln(1 - \delta) \right) - \sqrt{(1 + K)(m_1 c_1 + m_2 c_2)} \ln(1 - \delta) (\mu - \mu^+)
\]

\[
+ m(1 + K)(m_1 c_1 + 2m_2 c_2) (\mu - \mu^+) \ln \gamma^{-1}.
\]

(Recall that \( \gamma^{-1} = \mu^+ / \mu \geq \tau / \mu \).)

Since \( \tilde{\delta} \) and \( \gamma \) are constants, the lemma is established. \( \Box \)

Note that the previous lemma requires \( \tilde{\delta} < 1 \). However, evaluating \( \tilde{\delta} \) explicitly may not be possible. In the next lemma we will see that \( \tilde{\delta} \) is actually proportional to \( \delta \), which can be evaluated.

**Lemma 4.4.5.** For any \( \mu > 0 \) and \( x \in F_0 \), let \( \Delta x := -\{\nabla^2_{xx} \eta(\mu, x)\}^{-1} \nabla_x \eta(\mu, x) \) and \( \tilde{\Delta} x := x - x(\mu) \). We denote

\[
\delta := \delta(\mu, x) = \sqrt{-\frac{1}{\mu} \nabla^2_{xx} \eta(\mu, x)}[\Delta x, \Delta x] \quad \text{and} \quad \tilde{\delta} := \tilde{\delta}(\mu, x) = \sqrt{-\frac{1}{\mu} \nabla^2_{xx} \eta(\mu, x)}[\tilde{\Delta} x, \tilde{\Delta} x].
\]

If \( \delta < 1/6 \), then \( \frac{2}{3} \delta \leq \tilde{\delta} \leq 2 \delta \).

**Proof.** See the proof of Lemma 3.4.5. \( \Box \)

Theorem 4.4.2 follows by combining Lemmas 4.4.1(ii), 4.4.4, and 4.4.5.
Chapter 5

Some Applications

In this chapter of this dissertation, we will turn our attention to present four applications of SSOCPs and SRQCPs. Namely, we describe stochastic Euclidean facility location problem and the portfolio optimization problem with loss risk constraints as two applications of SSOCPs, then we describe the optimal covering random ellipsoid problem and an application in structural optimization as two applications of SRQCPs. We also refer the reader to a paper by Maggioni et al. [24] which describes another important application of SRQCPs in mobile ad hoc networks. The results of this chapter have been submitted for publication [2].

5.1 Two applications of SSOCPs

Each one of the following two subsections is devoted to describe an application of SSOCPs.

5.1.1 Stochastic Euclidean facility location problem

In facility location problems (FLPs) we are interested in choosing a location to build a new facility or locations to build multiple new facilities so that an appropriate measure of distance from the new facilities to existing facilities is minimized. FLPs arise locating
airports, regional campuses, wireless communication towers, etc.

The following are two ways of classifying FLPs (see also [39]): We can classify FLPs based on the number of new facilities in the following sense: if we add only one new facility then we get a problem known as a single facility location problem (SFLP), while if we add multiple new facilities instead of adding only one, then we get more a general problem known as a multiple facility location problem (MFLP). Another way of classification is based on the distance measure used in the model between the facilities. If we use the Euclidean distance then these problems are called Euclidean facility location problems (EFLPs), if we use the rectilinear distance then these problems are called rectilinear facility location problems (RFLPs).

In (deterministic) Euclidean single facility location problem (ESFLP), we are given \( r \) existing facilities represented by the fixed points \( a_1, a_2, \ldots, a_r \) in \( \mathbb{R}^n \), and we plan to place a new facility represented by \( x \) so that we minimize the weighted sum of the distances between \( x \) and each of the points \( a_1, a_2, \ldots, a_r \). This leads us to the problem

\[
\min \sum_{i=1}^{r} w_i \| x - a_i \|_2
\]

or, alternatively, to the problem (see for example [42])

\[
\min \sum_{i=1}^{r} w_i t_i \\
\text{s.t.} \quad (t_1; x - a_1; \ldots; t_r; x - a_r) \succeq_{(n+1)r} 0,
\]

where \( w_i \) is the weight associated with the \( i \)th existing facility and the new facility for \( i = 1, 2, \ldots, r \). However, the resulting model is a DSOCP model. So, there is no stochasticity in this model. But what if we assume that some of the fixed existing facilities are random? Where could such a randomness be found in the real world?

Picture (A) in Figure 5.1 shows the locations of multi-national military facilities led by troops from the United States and the United Kingdom in the Middle East region as
of December, 31, 2002, i.e., before the beginning of the Iraq war (it is known that the war began on March 20, 2003). This picture shows the multi-national military facilities including the navy facility locations and the air force facility locations but not the army facility locations. Assuming that the random existing facilities are the Iraqi military facilities whose locations are unknown at the time of taking this picture which is before the beginning of the war. Assuming also that the new facilities are the army multi-national facilities whose locations have to be determined at the time of taking this picture which is before the beginning of the war. Then it is reasonable to look at a problem of this kind as a stochastic ESFLP. Picture (B) in Figure 5.1 shows the locations of all multi-national military facility locations including the army facility locations as of December, 31, 2002, i.e., after starting the Iraq war.

So, in some applications, the locations of existing facilities cannot be fully specified because the locations of some of them depend on information not available at the time when decision needs to be made but will only be available at a later point in time. In general, in order to be precise only the latest information of the random facilities is used. This may require an increasing or decreasing of the number of the new facilities after
the latest information about the random facilities become available. In this case, we are interested in *stochastic facility location problems* (or abbreviated as stochastic FLPs). When the locations of all old facilities are fully specified, FLPs are called *deterministic facility location problems* (or abbreviated as deterministic FLPs).

In this section we consider (both single and multiple) stochastic Euclidean facility location problems, and in the next chapter we describe four different models of FLP, two of them can be viewed as generalizations of the models presented in this section.

**Stochastic ESFLPs**

Let \( a_1, a_2, \ldots, a_{r_1} \) be fixed points in \( \mathbb{R}^n \) representing the coordinates of \( r_1 \) existing fixed facilities and \( \tilde{a}_1(\omega), \tilde{a}_2(\omega), \ldots, \tilde{a}_{r_2}(\omega) \) be random points in \( \mathbb{R}^n \) representing the coordinates of \( r_2 \) random facilities who realizations depends on an underlying outcome \( \omega \) in an event space \( \Omega \) with a known probability function \( P \).

Suppose that at present we do not know the realizations of \( r_2 \) random facilities, and that at some point in time in future the realizations of these \( r_2 \) random facilities become known.

Our goal is to locate a new facility \( x \) that minimizes the weighted sums of the Euclidean distance between the new facility and each one of the existing fixed facilities and also minimizes the expected weighted sums of the distance between the new facility and the realization of each one of the random facilities. Note that this decision needs to be made before the realizations of the \( r_2 \) random facilities become available. This leads us to the following SSOCP model:

\[
\begin{align*}
\min & \quad \sum_{i=1}^{r_1} w_i t_i + \mathbb{E} \left[ Q(x, \omega) \right] \\
\text{s.t.} & \quad (t_1; x - a_1; \ldots; t_{r_1}; x - a_{r_1}) \preceq_{r_1} 0,
\end{align*}
\]

where \( Q(x, \omega) \) is the minimum value of the problem.
\[ \begin{align*} 
\min & \quad \sum_{j=1}^{r_1} \tilde{w}_j \bar{t}_j \\
\text{s.t.} & \quad (t_1; x - \tilde{a}_1(\omega); \ldots; t_{r_2}; x - \tilde{a}_{r_1}(\omega)) \succeq_{r_2} 0, \\
\end{align*} \]

and

\[ \mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega), \]

where \( w_i \) is the weight associated with the \( i \)th existing facility and the new facility for \( i = 1, 2, \ldots, r_1 \) and \( \tilde{w}_j(\omega) \) is the weight associated with the \( j \)th random existing facility and the new facility for \( j = 1, 2, \ldots, r_2 \).

**Stochastic EMFLPs**

Assume that we need to add \( m \) new facilities, namely \( x_1, x_2, \ldots, x_m \in \mathbb{R}^n \), instead of adding only one. This may require an increasing or decreasing of the number of the new facilities after the latest information about the random facilities become available. For simplicity, let us assume that the number of new facilities was previously known and fixed. Then we have two cases depending on whether or not there is an interaction among the new facilities in the underlying model. If there is no interaction between the new facilities, we are just concerned in minimizing the weighted sums of the distance between each one of the new facilities on one hand and each one of the fixed facilities and the realization of each one of the random facilities on the other hand. In other words, we solve the following SSOCP model:

\[ \begin{align*} 
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_1} w_{ij} t_{ij} + \mathbb{E} [Q(x_1; \ldots; x_m, \omega)] \\
\text{s.t.} & \quad (t_{ij}; x_j - a_1; \ldots; t_{r_1j}; x_j - a_{r_1}) \succeq_{r_1} 0, \quad j = 1, 2, \ldots, m, \\
\end{align*} \]

where \( Q(x_1; \ldots; x_m, \omega) \) is the minimum value of the problem.
\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_2} \tilde{w}_{ij}(\omega) \tilde{t}_{ij} \\
\text{s.t.} & \quad (\tilde{t}_{1j}; x_j - \tilde{a}_1(\omega); \ldots; \tilde{t}_{r_2j}; x_j - \tilde{a}_{r_2}(\omega)) \succeq_{r_2} 0, \quad j = 1, 2, \ldots, m,
\end{align*}
\]

and

\[
\mathbb{E}[Q(x_1; \ldots; x_m, \omega)] := \int_{\Omega} Q(x_1; \ldots; x_m, \omega) P(d\omega).
\]

where \( w_{ij} \) is the weight associated with the \( i \)-th existing facility and the \( j \)-th new facility for \( j = 1, 2, \ldots, m \) and \( i = 1, 2, \ldots, r_1 \), and \( \tilde{w}_{ij}(\omega) \) is the weight associated with the \( i \)-th random existing facility and the \( j \)-th new facility for \( j = 1, 2, \ldots, m \) and \( i = 1, 2, \ldots, r_2 \).

If interaction exists among the new facilities, then, in addition to the above requirements, we need to minimize the sum of the Euclidean distances between each pair of the new facilities. In this case, we are interested in a model of the form:

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_1} w_{ij} t_{ij} + \sum_{j=1}^{m} \sum_{j'=1}^{j-1} \tilde{w}_{jj'} \tilde{t}_{jj'} + \mathbb{E} [Q(x_1; \ldots; x_{r_1}, \omega)] \\
\text{s.t.} & \quad (t_{1j}; x_j - a_1(\omega); \ldots; t_{r_1j}; x_j - a_{r_1}) \succeq_{r_1} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\tilde{t}_{j(j+1)}; x_j - x_{j+1}; \ldots; \tilde{t}_{jm}; x_j - x_m) \succeq_{(m-j)} 0, \quad j = 1, 2, \ldots, m - 1,
\end{align*}
\]

where \( Q(x_1; \ldots; x_m, \omega) \) is the minimum value of the problem

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_2} \tilde{w}_{ij}(\omega) \tilde{t}_{ij} + \sum_{j=1}^{m} \sum_{j'=1}^{j-1} \tilde{w}_{jj'} \tilde{t}_{jj'} \\
\text{s.t.} & \quad (\tilde{t}_{1j}; x_j - \tilde{a}_1(\omega); \ldots; \tilde{t}_{r_2j}; x_j - \tilde{a}_{r_2}(\omega)) \succeq_{r_2} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\tilde{t}_{j(j+1)}; x_j - x_{j+1}; \ldots; \tilde{t}_{jm}; x_j - x_m) \succeq_{(m-j)} 0, \quad j = 1, 2, \ldots, m - 1,
\end{align*}
\]

and

\[
\mathbb{E}[Q(x_1; \ldots; x_m, \omega)] := \int_{\Omega} Q(x_1; \ldots; x_m, \omega) P(d\omega),
\]
where \( \hat{w}_{jj'} \) is the weight associated with the new facilities \( j' \) and \( j \) for \( j' = 1, 2, \ldots, j - 1 \) and \( j = 1, 2, \ldots, m \).

### 5.1.2 Portfolio optimization with loss risk constraints

The application in this subsection is a well-known problem from portfolio optimization. We consider the problem of maximizing the expected return subject to loss risk constraints. The same problem over one period was cited as an application of DSOCP (see Lobo, et al. [23]). Some extensions of this problem will be described.

**The problem**

Consider a portfolio problem with \( n \) assets or stocks over two periods. We start by letting \( x_i \) denote the amount of asset \( i \) held at the beginning of (and throughout) the first period, and \( p_i \) will denote the price change of asset \( i \) over this period. So, the vector \( p \in \mathbb{R}^n \) is the price vector over the first period. For simplicity, we let \( p \) be Gaussian with known mean \( \bar{p} \) and covariance \( \Sigma \), so the return over this period is the (scalar) Gaussian random variable \( r = p^T x \) with mean \( \bar{r} = \bar{p}^T x \) and variance \( \sigma = x^T \Sigma x \) where \( x = (x_1; x_2; \ldots; x_n) \).

Let \( y_i \) denote the amount of asset \( i \) held at the beginning of (and throughout) the second period, and \( q_i(\omega) \) will denote the random price change of asset \( i \) over this period whose realization depends on an underlying outcome \( \omega \) in an event space \( \Omega \) with known probability function \( P \). Similarly, we take the price vector \( q(\omega) \in \mathbb{R}^n \) over the second period to be Gaussian with mean \( \bar{q}(\omega) \) and covariance \( \bar{\Sigma}(\omega) \), so the return over this period is the (scalar) Gaussian random variable \( \tilde{r}(\omega) = q(\omega)^T y \) with mean \( \bar{\tilde{r}}(\omega) = \bar{q}(\omega)^T y \) and variance \( \tilde{\sigma}(\omega) = y^T \bar{\Sigma}(\omega) y \) where \( y = (y_1; y_2; \ldots; y_n) \). Suppose that in the first period we do not know the realization of the Gaussian vector \( q(\omega) \in \mathbb{R}^n \), and that at some point in time in the future (after the first period) the realization of this Gaussian vector becomes known. As pointed out in [23], the choices of portfolio variables \( x \) and \( y \) involve the
(classical, Markowitz) tradeoff between random return mean and random variance.

The optimization variables are the portfolio vectors \( \mathbf{x} \in \mathbb{R}^n \) (of the first stage) and \( \mathbf{y} \in \mathbb{R}^n \) (of the second stage). For these portfolio vectors, we take the simplest assumptions \( \mathbf{x} \geq 0, \mathbf{y} \geq 0 \) (i.e., no short positions [23]) and \( \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i = 1 \) (i.e. unit total budget [23]).

Let \( \alpha \) and \( \tilde{\alpha} \) be given unwanted return levels over the first and second periods, respectively, and let \( \beta \) and \( \tilde{\beta} \) be given maximum probabilities over the first and second periods, respectively. Assuming the above data is given, our goal is to determine the amount of the asset \( i \) (which is \( x_i \) over the first period and \( y_i \) over the second period), i.e. determine \( \mathbf{x} \) and \( \mathbf{y} \), such that the expected returns over these two periods are maximized subject to the following two types of loss risk constraints: the constraint \( P(r \leq \alpha) \leq \beta \) must be satisfied over the first period, and the constraint \( P(\tilde{r}(\omega) \leq \tilde{\alpha}) \leq \tilde{\beta} \) must be satisfied over the second period. This determination needs to be made before the realizations become available.

**Formulation of the model**

As noted in [23], the constraint \( P(r \leq \alpha) \leq \beta \) is equivalent to the second-order cone constraint

\[
\left( \alpha - \bar{r}; \Phi^{-1}(\beta)(\Sigma^{\frac{1}{2}} \mathbf{x}) \right) \succeq 0,
\]

provided \( \beta \leq 1/2 \) (i.e.,\( \Phi^{-1}(\beta) \leq 0 \)), where

\[
\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-t^2/2} dt
\]

is the cumulative normal distribution function of a zero mean unit variance Gaussian random variable. To prove this (see also [23]), notice that the constraint \( P(r \leq \alpha) \leq \beta \)
can be written as

\[ P \left( \frac{r - \bar{r}}{\sqrt{\sigma}} \leq \frac{\alpha - \bar{r}}{\sqrt{\sigma}} \right) \leq \beta. \]

Since \((r - \bar{r})/\sqrt{\sigma}\) is a zero mean unit variance Gaussian random variable, the probability above is simply \(\Phi((\alpha - \bar{r})/\sqrt{\sigma})\), thus the constraint \(P(r \leq \alpha) \leq \beta\) can be expressed as \(\Phi((\alpha - \bar{r})/\sqrt{\sigma}) \leq \beta\) or \((\alpha - \bar{r})/\sqrt{\sigma}) \leq \Phi^{-1}((\beta))\), or equivalently \(\bar{r} + \Phi^{-1}(\beta)\sqrt{\sigma} \geq \alpha\). Since 

\[
\sqrt{\sigma} = \sqrt{x^T \Sigma x} = \sqrt{(\Sigma^{1/2} x)^T (\Sigma^{1/2} x)} = \|\Sigma^{1/2} x\|_2,
\]

the constraint \(P(r \leq \alpha) \leq \beta\) is equivalent to the the second-order cone inequality \(\bar{r} + \Phi^{-1}(\beta)\|\Sigma^{1/2} x\|_2 \geq \alpha\) or equivalently to the second-order cone constraint 

\[
\begin{pmatrix} \alpha - \bar{r} ; \Phi^{-1}(\beta)(\Sigma^{1/2} x) \end{pmatrix} \succeq 0. 
\]

Similarly, provided \(\tilde{\beta} \leq 1/2\), the constraint \(P(\bar{r}(\omega) \leq \tilde{\alpha}) \leq \tilde{\beta}\) is equivalent to the second-order cone constraint 

\[
\begin{pmatrix} \tilde{\alpha} - \bar{r}(\omega) ; \Phi^{-1}(\beta)(\bar{\Sigma}(\omega)^{1/2} y) \end{pmatrix} \succeq 0. 
\]

Our goal is to determine the amount of the asset \(i\) (which is \(x_i\) over the first period and \(y_i\) over the second period), i.e. determine \(x\) and \(y\), such that the expected returns over these two periods are maximized. This problem can be cast as a two-stage SSOP as follows:

\[
\begin{align*}
\text{max} & \quad \tilde{p}^T x + \mathbb{E}[Q(x, \omega)] \\
\text{s.t.} & \quad \begin{pmatrix} \alpha - \tilde{p}^T x ; \Phi^{-1}(\beta)(\Sigma^{1/2} x) \end{pmatrix} \succeq 0 \\
& \quad 1^T x = 1, x \geq 0,
\end{align*}
\]

where \(Q(x, \omega)\) is the maximum value of the problem.
\[
\begin{align*}
\max & \quad \bar{q}(\omega)^T y \\
\text{s.t.} & \quad \left( \bar{\alpha} - \bar{q}(\omega)^T y; \Phi^{-1}(\bar{\beta})(\bar{\Sigma}(\omega)^{1/2} y) \right) \succeq 0 \\
& \quad 1^T y = 1, y \geq 0,
\end{align*}
\]

(5.1.2)

and

\[\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).\]

Note that this formulation is different from one suggested by Mehrotra and Özevin [27] which minimizes the variance period (two-stage extension of Markowitz’s mean-variance model). Here we formulate a model with a linear objective function leading to another approach for solving this problem.

**Extensions**

The simple problem described above has many extensions [23]. One of these extensions is imposing several loss risk constraints, i.e., the constraints \(P(r \leq \alpha_i) \leq \beta_i, \ i = 1, 2, \ldots, k_1\) (where \(\beta_i \leq 1/2, \ i = 1, 2, \ldots, k_1\)), or equivalently

\[
\left( \alpha_i - \bar{r}; \Phi^{-1}(\beta_i)(\Sigma^{1/2} x) \right) \succeq 0, \ \text{for} \ i = 1, 2, \ldots, k_1
\]

to be satisfied over the first period, and the constraints \(P(\bar{r}(\omega) \leq \bar{\alpha}_j) \leq \bar{\beta}_j, \ j = 1, 2, \ldots, k_2\) (where \(\bar{\beta}_j \leq 1/2, \ j = 1, 2, \ldots, k_2\)), or equivalently

\[
\left( \bar{\alpha}_j - \bar{r}(\omega); \Phi^{-1}(\bar{\beta}_j)(\bar{\Sigma}(\omega)^{1/2} y) \right) \succeq 0, \ \text{for} \ j = 1, 2, \ldots, k_2
\]

to be satisfied over the second period. So our problem becomes
\[
\max \ p^T x + \mathbb{E}[Q(x, \omega)] \\
\text{s.t.} \quad (\alpha_i - \bar{r}; \Phi^{-1}(\beta_i)(\Sigma^{1/2} x)) \succeq 0, \ i = 1, 2, \ldots, k_1 \\
1^T x = 1, x \geq 0,
\]

where \(Q(x, \omega)\) is the maximum value of the problem

\[
\max \ \bar{q}(\omega)^T y \\
\text{s.t.} \quad (\tilde{\alpha}_j - \bar{r}(\omega); \Phi^{-1}(\tilde{\beta}_j)(\tilde{\Sigma}(\omega)^{1/2} y)) \succeq 0, \ j = 1, 2, \ldots, k_2 \\
1^T y = 1, y \geq 0,
\]

and

\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).
\]

As another extension, observe that the statistical models \((\bar{p}; \Sigma)\) for the price changes during the first period and \(((\bar{q}(\omega), \tilde{\Sigma}(\omega))\) for the price changes during the second period are both uncertain (regardless of the fact that the later depends on \(\omega\) while the first does not) and one of the limitations of model in (5.1.1, 5.1.2) is its need to estimate these statistical models. In [11], Bawa et al. indicated that using estimates of unknown expected returns and unknown covariance matrices leads to an estimation risk in portfolio choice. To handle this uncertainty, as in [11], we take those expected returns and covariance matrices that belong to bounded intervals. i.e., we consider the statistical models \((\bar{p}; \Sigma)\) and \(((\bar{q}(\omega), \tilde{\Sigma}(\omega))\) that belong to the bounded sets:

\[
\mathcal{B} := \{(\bar{p}; \Sigma) : \bar{p}_l \leq \bar{p} \leq \bar{p}_u, \Sigma_l \leq \Sigma \leq \Sigma_u\}\]
and
\[ R := \{ (\bar{q}(\omega); \bar{\Sigma}(\omega)) : \bar{q}_l(\omega) \leq q(\omega) \leq \bar{q}_u(\omega), \bar{\Sigma}_l(\omega) \leq \bar{\Sigma}(\omega) \leq \bar{\Sigma}_u(\omega) \}, \]

respectively, where \( \bar{p}_l, \bar{p}_u, \bar{q}_l(\omega), \bar{q}_u(\omega), \Sigma_l, \Sigma_u, \bar{\Sigma}_l(\omega) \) and \( \bar{\Sigma}_u(\omega) \) are the extreme values of the intervals mentioned above, then we take a worst case realization of the statistical models \( (\bar{p}; \Sigma) \) and \( (\bar{q}(\omega), \bar{\Sigma}(\omega)) \) by maximizing the minimum of the expected returns over all \( (\bar{p}; \Sigma) \in \mathcal{R} \) and \( (\bar{q}(\omega), \bar{\Sigma}(\omega)) \in R \).

Let us consider here only a discrete version of this development. Suppose we have \( N_1 \) different possible scenarios over the first period, each of which is modeled by a simple Gaussian model for the price change vector \( p_k \) with mean \( \bar{p}_k \) and covariance \( \Sigma_k \). Similarly, we also have \( N_2 \) different possible scenarios over the second period, each of which is modeled by a simple Gaussian model for the price change \( q_l(\omega) \), with mean \( \bar{q}_l(\omega) \) and covariance \( \Sigma_l(\omega) \). We can then take a worst case realization of these information by maximizing the minimum of the expected returns for these different scenarios, subject to a constraint on the loss risk for each scenario to get the following SSOCP model:

\[
\begin{align*}
\max \quad & \bar{p}^T x + \mathbb{E}[Q(x, \omega)] \\
\text{s.t.} \quad & \left( \alpha - \bar{p}_i^T x; \Phi^{-1}(\beta)(\Sigma_i^{1/2} x) \right) \succeq 0, \quad i = 1, 2, \ldots, N_1 \\
& 1^T x = 1, \quad x \geq 0,
\end{align*}
\]

where \( \mathbb{E}[Q(x, \omega)] \) is the maximum value of the problem

\[
\begin{align*}
\max \quad & \bar{q}(\omega)^T y \\
\text{s.t.} \quad & \left( \hat{\alpha} - \bar{q}_j(\omega)^T y; \Phi^{-1}(\hat{\beta})(\bar{\Sigma}_j(\omega)^{1/2} y) \right) \succeq 0, \quad j = 1, 2, \ldots, N_2 \\
& 1^T y = 1, \quad y \geq 0,
\end{align*}
\]

and

\[ \mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega). \]
5.2 Two applications of SRQCPs

The stochastic versions of the two problems in this section have been described and formulated by Ariyawansa and Zhu [49] as SSDP models, but it has been found recently [1, 23, 24] that on numerical grounds solving DSOCPs (SSOCPs) or DRQCPs (SRQCPs) by treating them as DSDPs (SSDPs) is inefficient, and that DSOCPs (SSOCPs) or DRQCPs (SRQCPs) can be solved more efficiently by exploiting their structure. In fact, in [1, 23] we see many problems first formulated as DSDPs in [38, 41, 30] have been reformulated as DSOCPs or DRQCP mainly for this reason. In view of this fact, in this section we reformulate SRQCP models (as these problems should be solved as such) for the minimum-volume covering random ellipsoid problem and the structural optimization problem.

5.2.1 Optimal covering random ellipsoid problem

The problem

The problem description in this part is taken from Ariyawansa and Zhu [9, Subsection 3.2]. Suppose we have \( n_1 \) fixed ellipsoids:

\[
\mathcal{E}_i := \{ x \in \mathbb{R}^n : x^T H_i x + 2 g_i^T x + v_i \leq 0 \} \subset \mathbb{R}^n, i = 1, 2, \ldots, n_1
\]

where \( H_i \in S^n_+, g_i \in \mathbb{R}^n \) and \( v_i \in \mathbb{R} \) are deterministic data for \( i = 1, 2, \ldots, n_1 \).

Suppose we also have \( n_2 \) random ellipsoids:

\[
\tilde{\mathcal{E}}_i(\omega) := \{ x \in \mathbb{R}^n : x^T \tilde{H}_i(\omega) x + 2 \tilde{g}_i(\omega)^T x + \tilde{v}_i(\omega) \leq 0 \} \subset \mathbb{R}^n, i = 1, 2, \ldots, n_2,
\]

where, for \( i = 1, 2, \ldots, n_2, \tilde{H}_i(\omega) \in S^n_+, \tilde{g}_i(\omega) \in \mathbb{R}^n \) and \( \tilde{v}_i(\omega) \in \mathbb{R} \) are random data whose realizations depend on an underlying outcome \( \omega \) in an event space \( \Omega \) with a known

---

1The acronym DRQCPs stands for deterministic rotated quadratic cone programs.
probability function $P$. We assume that, at present time, we do not know the realizations of $n_2$ random ellipsoids, and that at some point in the future the realizations of these $n_2$ ellipsoids become known.

Given this, we need to determine a ball that contains all $n_2$ fixed ellipsoids and the realizations of the $n_2$ random ellipsoids. This decision needs to be made before the realizations of the random ellipsoids become available. Consequently, when the realizations of the random ellipsoids do become available, the ball that has already been determined may or may not contain all the realized random ellipsoids. In order to guarantee that the modified ball contains all (fixed and realizations of random) ellipsoids, we assume that at that stage we are allowed to change the radius of the ball (but not its center), if necessary.

We consider the same assumptions as in [49]. We assume that the cost of choosing the ball has three components:

- the cost of the center, which is proportional to the Euclidean distance to the center from the origin;
- the cost of the initial radius, which is proportional to the square of the radius;
- the cost of changing the radius. The center and the radius of the initial ball are determined so that the expected total cost is minimized.

In [49], Ariyawansa and Zhu describe the following concrete version of this generic application: Let $n := 2$. The fixed ellipsoids contain targets that need to be destroyed, and the random ellipsoids contain targets that also need to be destroyed but are moving. Fighter aircrafts take off from the origin with a planned disk of coverage that contains the fixed ellipsoids. In order to be accurate only the latest information about the moving targets is used. This may require increasing the radius of the planned disk of coverage after latest information about the random ellipsoids become available, which may occur after the initially planned fighter aircrafts have taken off. This increase, dependent on
the initial disk of coverage and the specific information about the moving targets, may result in an additional cost. The initial disk of coverage need to be determined so that the expected total cost is minimized.

Our first goal is to determine \( \bar{x} \in \mathbb{R}^n \) and \( \gamma \in \mathbb{R} \) such that the ball \( B \) defined by

\[
B := \{ x \in \mathbb{R}^n : x^T x - 2\bar{x}^T x + \gamma \leq 0 \}
\]

contains the fixed ellipsoids \( E_i \) for \( i = 1, 2, \ldots, n_1 \). As we mentioned, this determination need to be determined before the realization of the random ellipsoids become available. When the realizations of the random ellipsoids become available, if necessary, we need to determine \( \tilde{\gamma} \) so that the new ball

\[
\tilde{B} := \{ x \in \mathbb{R}^n : x^T x - 2\bar{x}^T x + \tilde{\gamma} \leq 0 \}
\]

contains all the realizations of the random ellipsoids \( \tilde{E}_i(\omega) \) for \( i = 1, 2, \ldots, n_2 \).

Notice that the center of the ball \( B \) is \( \bar{x} \), its radius is \( r := \sqrt{\bar{x}^T \bar{x} - \gamma} \), and the distance from the origin to its center is \( \sqrt{\bar{x}^T \bar{x}} \). Notice also that the new ball \( \tilde{B} \) has the same center \( \bar{x} \) as \( B \) but a larger radius \( \tilde{r} := \sqrt{\bar{x}^T \bar{x} - \tilde{\gamma}} \).

**Formulation of the model**

We introduce the constraints \( d_1^2 \geq \bar{x}^T \bar{x} \) and \( d_2 \geq r^2 = \bar{x}^T \bar{x} - \gamma \). That is, \( d_1 \) is an upper bound on the distance between the center of the ball \( B \) and the origin, \( \sqrt{\bar{x}^T \bar{x}} \), and \( d_2 \) is an upper bound on square of the radius of the ball \( B \).

In order to proceed, we feel that it is necessary for the reader to recall the following fact:

**Fact 1** (Sun and Freund [37]). Suppose that we are given two ellipsoids \( E_i \subset \mathbb{R}^n, i = 1, 2 \) defined by \( E_i := \{ x \in \mathbb{R}^n : x^T H_i x + 2g_i^T x + v_i \leq 0 \} \), where \( H_i \in S^n_+ \), \( g_i \in \mathbb{R}^n \) and \( v_i \in \mathbb{R} \).
for \( i = 1, 2 \), then \( \mathcal{E}_1 \) contains \( \mathcal{E}_2 \) if and only if there exists \( \tau \geq 0 \) such that the linear matrix inequality
\[
\begin{bmatrix}
H_1 & g_1 \\
g_1^T & v_1
\end{bmatrix} \preceq \tau
\begin{bmatrix}
H_2 & g_2 \\
g_2^T & v_2
\end{bmatrix}
\]
holds.

In view of Fact 1 and the requirement that the ball \( \mathcal{B} \) contains the fixed ellipsoids \( \mathcal{E}_i \) for \( i = 1, 2, \ldots, n_1 \), and the realizations of the random ellipsoids \( \tilde{\mathcal{E}}_i(\omega) \) for \( i = 1, 2, \ldots, n_2 \), we accordingly add the following constraints:

\[
\begin{bmatrix}
I & -\bar{x} \\
-\bar{x}^T & \gamma
\end{bmatrix} \preceq \tau_i
\begin{bmatrix}
H_i & g_i \\
g_i^T & v_i
\end{bmatrix}, \quad i = 1, 2, \ldots, n_1,
\]

and

\[
\begin{bmatrix}
I & -\bar{x} \\
-\bar{x}^T & \tilde{\gamma}
\end{bmatrix} \preceq \delta_i
\begin{bmatrix}
\tilde{H}_i(\omega) & \tilde{g}_i(\omega) \\
\tilde{g}_i^T(\omega) & v_i(\omega)
\end{bmatrix}, \quad i = 1, 2, \ldots, n_2,
\]

or equivalently

\[
M_i \succeq 0, \quad \forall i = 1, \ldots, n_1 \quad \text{and} \quad \tilde{M}_i(\omega) \succeq 0, \quad i = 1, 2, \ldots, n_2
\]

where for each \( i = 1, \ldots, n_1 \),

\[
M_i := \begin{bmatrix}
\tau_i H_i - I & \tau_i g_i + \bar{x} \\
\tau_i g_i^T + \bar{x}^T & \tau_i v_i - \gamma
\end{bmatrix},
\]

and for each \( i = 1, \ldots, n_2 \),

\[
\tilde{M}_i(\omega) := \begin{bmatrix}
\delta_i \tilde{H}_i(\omega) - I & \delta_i \tilde{g}_i(\omega) + \bar{x} \\
\delta_i \tilde{g}_i^T(\omega) + \bar{x}^T & \delta_i v_i(\omega) - \tilde{\gamma}
\end{bmatrix}.
\]
Since we are looking to minimizing $d_2$, where $d_2$ is an upper bound on square of the radius of the ball $B$, we can write the constraint $d_2 \geq \bar{x}^T \bar{x} - \gamma$ as $d_2 = \bar{x}^T \bar{x} - \gamma$. So, the matrix $M_i$ can be then written as

$$M_i = \begin{bmatrix} \tau_i H_i - I & \tau_i g_i + \bar{x} \\ \tau_i g_i^T + \bar{x}^T & \tau_i v_i + d_2 - \bar{x}^T \bar{x} \end{bmatrix}.$$ 

Now, let $H_i := \Xi_i \Lambda_i \Xi_i^T$ be the spectral decomposition of $H_i$, where $\Lambda_i := \text{diag}(\lambda_{i1}; \ldots; \lambda_{in})$, and let $u_i := \Xi_i^T (\tau_i g_i + \bar{x})$. Then, for each $i = 1, \ldots, n_1$, we have

$$\bar{M}_i := \begin{bmatrix} \Xi_i^T & 0 \\ 0^T & 1 \end{bmatrix} M_i \begin{bmatrix} \Xi_i & 0 \\ 0^T & 1 \end{bmatrix} = \begin{bmatrix} \tau_i \Lambda_i - I & u_i \\ u_i^T & \tau_i v_i + d_2 - \bar{x}^T \bar{x} \end{bmatrix}.$$ 

Consequently, $M_i \succeq 0$ if and only if $\bar{M}_i \succeq 0$ for each $i = 1, 2, \ldots, n_1$. Now our formulation of the problem in SSOCP depends on the following lemma (see also [23]): 

Lemma 5.2.1. For each $i = 1, 2, \ldots, n_1$, $\bar{M}_i \succeq 0$ if and only if $\tau_i \lambda_{\min}(H_i) \geq 1$ and $\bar{x}^T \bar{x} \leq d_2 + \tau_i v_i + 1^T s_i$, where $s_i = (s_{i1}; \ldots; s_{in})$, $s_{ij} = u_{ij}^2 / (\tau_i \lambda_{ij} - 1)$ for all $j$ such that $\tau_i \lambda_{ij} > 1$, and $s_{ij} = 0$ otherwise.

Proof. For each $i = 1, 2, \ldots, n_1$, it is known that the matrix $\bar{M}_i \succeq 0$ if and only if every principle minor of $\bar{M}_i$ is nonnegative. Since

$$\det(\tau_i \Lambda_i - I) = \begin{vmatrix} \tau_i \lambda_{i1} - 1 & 0 & \ldots & 0 \\ 0 & \tau_i \lambda_{i2} - 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \tau_{in} \lambda_{in} - 1 \end{vmatrix} = \Pi_{j=1}^{n_1} (\tau_j \lambda_{ij} - 1).$$

It follows that $\bar{M}_i \succeq 0$ if and only if $\Pi_{j=1}^{n_1} (\tau_j \lambda_{ij} - 1) \geq 0$, for all $s = 1, \ldots, n_1$ and $\det(\bar{M}_i) \geq 0$. Thus, $\bar{M}_i \succeq 0$ if and only if $\tau_j \lambda_{ij} \geq 1$, for all $j = 1, \ldots, n_1$ and $\det(\bar{M}_i) \geq 0$. 

122
Notice that

\[ \det(M_i) = \left( \prod_{j=1}^{k} (\tau_i \lambda_{ij} - 1) \right) \left( (\tau_i v_i + d_2 - \bar{x}^T \bar{x}) - \sum_{j=1}^{k} u_{ij}^2 \right). \]

This means the inequality \( \det(M_i) \geq 0 \) strictly holds for each \( i \leq j \leq k \) such that \( \tau_j \lambda_{ij} = 1 \). Hence, \( \det(M_i) \geq 0 \) if and only if

\[ (\tau_i v_i + d_2 - \bar{x}^T \bar{x}) - \sum_{\tau_i \lambda_{ij} > 1} (u_{ij}^2 / (\tau_j \lambda_{ij} - 1)) \geq 0. \]

Therefore, \( \tilde{M}_i \geq 0 \) if and only if \( \tau_i \lambda_{\min}(H_i) \geq 1 \) and \( d_2 \geq \bar{x}^T \bar{x} - \tau_i v_i + 1^T s_i \). \( \Box \)

So far, we have shown that each constraint \( \tilde{M}_i \geq 0 \) can be replaced by the constraints \( \tau_i \lambda_{\min}(H_i) \geq 1, \bar{x}^T \bar{x} \leq \sigma \) and \( \sigma \leq d_2 + \tau_i v_i - 1^T s_i \).

Similarly, for each \( i = 1, 2, \ldots, n \), by letting \( \tilde{H}_i := \tilde{\Xi}_i^T \tilde{\Lambda}_i \tilde{\Xi}_i^T \) (the spectral decomposition of \( \tilde{H}_i \)), \( \tilde{u}_i := \tilde{\Xi}_i^T (\delta_i \tilde{g}_i + \bar{x}) \), and \( \tilde{s}_i := (\tilde{s}_{i1}; \tilde{s}_{i2}; \ldots; \tilde{s}_{in}) \), where \( \tilde{s}_{ij} := \tilde{u}_{ij}^2 / (\delta_i \tilde{\lambda}_{ij} - 1) \) for all \( j \) when \( \delta_i \tilde{\lambda}_{ij} > 1 \), and \( \tilde{s}_{ij} := 0 \) otherwise, then we can show that each constraint \( \tilde{M}_i(\omega) \geq 0 \) can be replaced by the constraints \( \delta_i \tilde{\lambda}_{\min}(\tilde{H}_i(\omega)) \geq 1, \bar{x}^T \bar{x} \leq \tilde{\sigma}(\omega) \), and \( \tilde{\sigma}(\omega) \leq \tilde{d}_2 + \delta_i v_i(\omega) - 1^T \tilde{s}_i(\omega) \).

Since we are minimizing \( d_2 \) and \( \tilde{d}_2 \), then for all \( j = 1, 2, \ldots, n \), we can relax the definitions of \( s_{ij} \) and \( \tilde{s}_{ij} \) by replacing them by \( u_{ij}^2 \leq s_{ij}(\tau_i \lambda_{ij} - 1) \) for all \( i = 1, 2, \ldots, n_1 \) and \( \tilde{u}_{ij}^2 \leq \tilde{s}_{ij}(\delta_i \tilde{\lambda}_{ij} - 1) \) for all \( i = 1, 2, \ldots, n_2 \), respectively.

When the realizations of the random ellipsoids become available, if necessary, we determine \( \tilde{\lambda} \) so that the new ball

\[ \tilde{B} := \{ x \in \mathbb{R}^n : x^T x - 2 \bar{x}^T x + \tilde{\lambda} \leq 0 \} \]

contains all the realizations of the random ellipsoid. This new ball \( \tilde{B} \) has the same center \( \bar{x} \) as \( B \) but a larger radius, \( \tilde{r} := \sqrt{\bar{x}^T \bar{x}} \). We note that \( \tilde{r}^2 - r^2 = (\bar{x}^T \bar{x} - \gamma) - (\bar{x}^T \bar{x} - \gamma) = \gamma - \gamma, \)
and thus we introduce the constraint $0 \leq \gamma - \tilde{\gamma} \leq z$ where $z$ is an upper bound of $r^2 - \tilde{r}^2$.

Let $\bar{c} > 0$ denote the cost per unit of the Euclidean distance between the center of the ball $\mathcal{B}$ and the origin; let $\alpha > 0$ be the cost per unit of the square of the radius of $\mathcal{B}$; and let $\beta > 0$ be the cost per unit increase of the square of the radius if it becomes necessary after the realizations of the random ellipsoids are available.

We now define the following decision variables

$$x := (d_1; d_2; \bar{x}; \gamma; \tau) \quad \text{and} \quad y := (z; \tilde{\gamma}; \delta).$$

Then, by introducing the following unit cost vectors

$$c := (\bar{c}; \alpha; 0; 0) \quad \text{and} \quad q := (\beta; 0; 0),$$

and combining all of the above, we get the following SQRCP model:

$$\min \quad c^T x + \mathbb{E}[Q(x, \omega)]$$

s.t. \quad $u_i = \Xi_i^T (\tau_i g_i + \bar{x}), i = 1, 2, \ldots, n_1$

\quad $u_{ij}^2 \leq s_{ij}(\tau_i \lambda_{ij} - 1), i = 1, 2, \ldots, n_1, j = 1, 2, \ldots, n$

\quad $\bar{x}^T \bar{x} \leq \sigma_1$

\quad $\sigma_1 \leq d_2 + \tau_i v_i - 1^T s_i, i = 1, 2, \ldots, n_1$

\quad $\tau_i \geq 1/\lambda_{\min}(H_i), i = 1, 2, \ldots, n_1$

\quad $\bar{x}^T \bar{x} \leq d_1^2$,

where $Q(x, \omega)$ is the minimum value of the problem.
\[
\begin{align*}
\min & \quad q^T y \\
\text{s.t.} & \quad \bar{u}_i(\omega) = \tilde{Z}_i^T(\omega)(\delta_i \tilde{g}_i(\omega) + \bar{x}), i = 1, 2, \ldots, n_2 \\
& \quad \tilde{u}_{ij}(\omega)^2 \leq \tilde{s}_{ij}(\omega) (\delta_i \tilde{\lambda}_{ij}(\omega) - 1), i = 1, 2, \ldots, n_2, j = 1, 2, \ldots, n \\
& \quad \bar{x}^T \bar{x} \leq \tilde{\sigma}_2(\omega) \\
& \quad \tilde{\sigma}_2(\omega) \leq d_2 + \delta_i v_i(\omega) - 1^T \tilde{s}_i(\omega), i = 1, 2, \ldots, n_2 \\
& \quad \delta_i \geq 1/\tilde{\lambda}_{\min}(\tilde{H}_i)(\omega), i = 1, 2, \ldots, n_2 \\
& \quad 0 \leq \delta - \tilde{\delta} \leq z,
\end{align*}
\]

and
\[
E[Q(x, \omega)] := \int_\Omega Q(x, \omega) P(d\omega).
\]

### 5.2.2 Structural optimization

Ben-Tal and Bends\o e in [12] and Nemirovski [13] consider the following problem from structural optimization. A structure of \( k \) linear elastic bars connects a set of \( p \) nodes. They assume the geometry (topology and lengths of bars) and the material are fixed. The goal is to size the bars, i.e., determine appropriate cross-sectional areas of the bars.

For \( i = 1, 2, \ldots, k \), and \( j = 1, 2, \ldots, p \), we define the following decision variables and parameters:

- \( f_j := \) the external force applied on the \( j^{th} \) node,
- \( d_j := \) the (small) displacement of the \( j^{th} \) node resulting from the load force \( f_j \),
- \( x_i := \) the cross-sectional area of the \( i^{th} \) bar,
- \( \underline{x}_i := \) the lower bound on the cross-sectional area of the \( i^{th} \) bar,
- \( \overline{x}_i := \) the upper bound on the cross-sectional area of the \( i^{th} \) bar,
- \( l_i := \) the length of the \( i^{th} \) bar,
• \( v := \) the maximum allowed volume of the bars of the structure,

• \( G(x) := \sum_{i=1}^{k} x_i G_i \) is the stiffness matrix, where the matrices \( G_i \in S^p, i = 1, 2, \ldots, k \) depend only on fixed parameters (such as length of bars and material).

In the simplest version of the problem they consider one fixed set of externally applied nodal forces \( f_j, j = 1, 2, \ldots, p \). Given this, the elastic stored energy within the structure is given by

\[
\varepsilon = f^T d,
\]

which is a measure of the inverse of the stiffness of the structure. In view of the definition of the stiffness matrix \( G(x) \), we can also conclude that the following linear relationship between \( f \) and \( d \):

\[
f = G(x) \ d.
\]

The objective is to find the stiffest truss by minimizing \( \varepsilon \) subject to the inequality \( l^T x \leq v \) as a constraint on the total volume (or equivalently, weight) and the constraint \( \underline{x} \leq x \leq \bar{x} \) as upper and lower bounds on the cross-sectional areas.

For simplicity, we assume that \( x > 0 \) and \( G(x) \succ 0 \), for all \( x > 0 \). In this case we can express the elastic stored energy in terms of the inverse of the stiffness matrix and the external applied nodal force as follows:

\[
\varepsilon = f^T G(x)^{-1} f.
\]

In summary, they consider the problem

\[
\begin{align*}
\min \quad & f^T G(x)^{-1} f \\
\text{s.t.} \quad & \underline{x} \leq x \leq \bar{x} \\
& l^T x \leq v,
\end{align*}
\]
which is equivalent to

$$\begin{align*}
\min & \quad s \\
\text{s.t.} & \quad f^T G(x)^{-1} f \leq s \\
& \quad x \leq x \leq \bar{x} \\
& \quad l^T x \leq v.
\end{align*}$$

(5.2.1)

The first inequality constraint in (5.2.1) is just fractional quadratic function inequality constraint and it can be formulated as a hyperbolic inequality. In §2 of [1] (see also §2 of [23]), the authors demonstrate that this inequality is satisfied if and only if there exists $u_i \in \mathbb{R}^{r_i}$ and $t_i \in \mathbb{R}$ with $t_i > 0$, $i = 1, 2, \ldots, k$, such that

$$\sum_{i=1}^{k} D_i^T u_i = f, \quad u_i^T u_i \leq x_i t_i, \text{ for } i = 1, 2, \ldots, k, \text{ and } 1^T t \leq s,$$

where $r_i = \text{rank}(G_i)$ and the matrix $D_i \in \mathbb{R}^{r_i \times n}$ is the Cholesky factor of $G_i$ (i.e., $D_i^T D_i = G_i$) for each $i = 1, 2, \ldots, k$. Using this result, problem (5.2.1) becomes the problem

$$\begin{align*}
\min & \quad s \\
\text{s.t.} & \quad \sum_{i=1}^{k} D_i^T u_i = f \\
& \quad u_i^T u_i \leq x_i t_i, \text{ for } i = 1, \ldots, k \\
& \quad 1^T t \leq s \\
& \quad x \leq x \leq \bar{x} \\
& \quad l^T x \leq v.
\end{align*}$$

(5.2.2)

which includes only linear and hyperbolic constraints.

The model (5.2.2) depends on a “simple” assumption that says that the external forces applied to the nodes are fixed. As more complicated versions, they consider multiple loading scenarios as well. In [49], Ariyawanza and Zhu consider the case that external forces applied to the nodes are random variables with known distribution functions and they formulate an SSDP model for this problem. In this subsection, we formulate an
SRQCP model for this problem under the assumption that some of the external forces applied to the nodes are fixed and the rest are random variables with known distribution functions. Let us denote to the external force applied on the \(j^{th}\) node by \(\tilde{f}_j\) if it is fixed and by \(\tilde{f}_j(\omega)\) if it is random. Due to the changes in the environmental conditions (such as wind speed and temperature), we believe that the randomness of some of the external forces is much closer to the reality. Without loss of generality, let us assume that \(f(\omega) = (\tilde{f}; \tilde{f}(\omega))\) where \(\tilde{f}(\omega)\) depends on an underlying outcome \(\omega\) in an event space \(\Omega\) with a known probability function \(P\).

Accordingly, the displacement of the \(j^{th}\) node resulting from the random forces and the elastic stored energy within the structure are also random. Then we have the following relations

\[
\tilde{\epsilon} = \tilde{f}^T \tilde{d}, \quad \tilde{f} = G(x) \tilde{d}, \quad \tilde{\epsilon}(\omega) = \tilde{f}(\omega)^T \tilde{d}(\omega), \quad \text{and} \quad \tilde{f}(\omega) = G(x) \tilde{d}(\omega).
\]

Suppose we design a structure for a customer. The structure will be installed in an open environment. From past experience, the customer can provide us with sufficient information so that we can model the external forces that will be applied to the nodes of the structure as random variables with known distribution functions. Given this information, we can formulate a model of this problem so that we can guarantee that our structure will be able to continue to function and stand up against the worst environmental conditions.
In summary, we solve the following SRQCP problem:

\[
\begin{align*}
\min & \quad \tilde{s} + \mathbb{E} [Q(x, \omega)] \\
\text{s.t.} & \quad \sum_{i=1}^{k} D_i^T \hat{u}_i = \hat{f} \\
& \quad \hat{u}_i^T \hat{u}_i \leq x_i \hat{t}_i, \quad i = 1, \ldots, k \\
& \quad \mathbf{1}^T \hat{t} \leq \tilde{s} \\
& \quad x \leq x \leq \bar{x} \\
& \quad l^T x \leq v,
\end{align*}
\]

where \( Q(x, \omega) \) is the minimum value of the problem

\[
\begin{align*}
\min & \quad \tilde{s} \\
\text{s.t.} & \quad \sum_{i=1}^{k} D_i^T \tilde{u}(\omega) = \tilde{f}(\omega) \\
& \quad \tilde{u}(\omega)^T \tilde{u}(\omega) \leq x_i \tilde{t}_i, \quad i = 1, \ldots, k \\
& \quad \mathbf{1}^T \tilde{t} \leq \tilde{s},
\end{align*}
\]

and

\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).
\]
Chapter 6

Related Open problems:

Multi-Order Cone Programming Problems

In this chapter we introduce a new class of convex optimization problems that can be viewed as an extension of second-order cone programs. We present primal and dual forms of multi-order cone programs (MOCPs) in which we minimize a linear function over a Cartesian product of $p^{th}$-order cones (we allow different $p$ values for different cones in the product). We then indicate weak and strong duality relations for the problem. We also introduce mixed integer multi-order cone programs (MIMOCPs) to handle MOCPs with integer-valued variables, and two-stage stochastic multi-order cone programs (SMOCPs) with recourse to handle uncertainty in data defining (deterministic) MOCPs. We demonstrate how decision making problems associated with facility location problems lead to MOCP, MIMOCP, and SMOCP models. It is interesting to investigate other applicational settings leading to (deterministic, stochastic and mixed integer) MOCPs. Development of algorithms for such multi-order cone programs which in turn will benefit from a duality theory is equally interesting and remains one of the important open problems for future
We begin by introducing some notations we use in the sequel.

Given $p \geq 1$, the $p$th-order cone of dimension $n$ is defined as

$$Q^n_p := \{x = (x_0; \bar{x}) \in \mathbb{R} \times \mathbb{R}^{n-1} : x_0 \geq \|\bar{x}\|_p\}$$

where $\| \cdot \|_p$ denotes the $p$-norm. The cone $Q^n_p$ is regular (see, for example, [45]). As special cases, when $p = 2$ we obtain $Q^n_2 := E_n^+; \text{ the second-order cone of dimension } n$, and when $p = 1$ or $\infty$, $Q^n_p$ is a polyhedral cone.

We write $x \succeq^{(n)}_{(p)} 0$ to mean that $x \in Q^n_p$. Given $1 \leq p_i \leq \infty$ for $i = 1, 2, \ldots, r$. We write $x \succeq^{(n_1,n_2,\ldots,n_r)}_{(p_1,p_2,\ldots,p_r)} 0$ to mean that $x \in Q^{n_1}_{p_1} \times Q^{n_2}_{p_2} \times \cdots \times Q^{n_r}_{p_r}$. It is immediately seen that, for every vector $x \in \mathbb{R}^n$ where $n = \sum_{i=1}^{r} n_i$, $x \succeq^{(n_1,n_2,\ldots,n_r)}_{(p_1,p_2,\ldots,p_r)} 0$ if and only if $x$ is partitioned conformally as $x = (x_1; x_2; \ldots; x_r)$ and $x_i \succeq^{(n_i)}_{(p_i)} 0$ for $i = 1, 2, \ldots, r$. For simplicity, we write:

- $Q^n_p$ as $Q_p$ and $x \succeq^{(n)}_{(p)} 0$ as $x \succeq^{(n)}_{(p)} 0$ when $n$ is known from the context;
- $Q^{n_1}_{p_1} \times Q^{n_2}_{p_2} \times \cdots \times Q^{n_r}_{p_r}$ as $Q_{(p_1,p_2,\ldots,p_r)}$ and $x \succeq^{(n_1,n_2,\ldots,n_r)}_{(p_1,p_2,\ldots,p_r)} 0$ as $x \succeq^{(n_1,n_2,\ldots,n_r)}_{(p_1,p_2,\ldots,p_r)} 0$ when $n_1, n_2, \ldots, n_r$ are known from the context;
- $x \succeq^{(p_1,p_2,\ldots,p_r)}_{\text{r-times}} 0$ as $x \succeq^{(p_1,p_2,\ldots,p_r)}_{\text{r-times}} 0$.

The set of all interior points of $Q^n_p$ is denoted by $\text{int}(Q^n_p) := \{x = (x_0; \bar{x}) \in \mathbb{R} \times \mathbb{R}^{n-1} : x_0 > \|\bar{x}\|_p\}$. We write $x \succ^{(p_1,p_2,\ldots,p_r)} 0$ to mean that $x \in \text{int}(Q_{(p_1,p_2,\ldots,p_r)}) := \text{int}(Q_{p_1}) \times \text{int}(Q_{p_2}) \times \cdots \times \text{int}(Q_{p_r})$.

### 6.1 Multi-order cone programming problems

Let $r \geq 1$ be an integer, and $p_1, p_2, \ldots, p_r$ are such that $1 \leq p_i \leq \infty$ for $i = 1, 2, \ldots, r$. Let $m, n, n_1, n_2, \ldots, n_r$ be positive integers such that $n = \sum_{i=1}^{r} n_i$. Then we define an
MOCP in *primal standard form* as

\[
\min \ c^T x \\
\text{s.t.} \ A x = b \\
x \succeq_{(p_1,p_2,\ldots,p_r)} 0,
\]

where \( A \in \mathbb{R}^{m \times n}, \ b \in \mathbb{R}^m \) and \( c \in \mathbb{R}^n \) constitute given data, and \( x \in \mathbb{R}^n \) is the primal decision variable. We define a MOCP in *dual standard form* as

\[
\max \ b^T y \\
\text{s.t.} \ A^T y + z = c \\
z \succeq_{(q_1,q_2,\ldots,q_r)} 0,
\]

where \( y \in \mathbb{R}^m \) and \( z \in \mathbb{R}^n \) are the dual decision variables, and \( q_1, q_2, \ldots, q_r \) are integers such that \( 1 \leq q_i \leq \infty \) for \( i = 1, 2, \ldots, r \).

If (P) and (D) are defined by the same data, and \( q_i \) is conjugate to \( p_i \), in the sense that \( 1/p_i + 1/q_i = 1 \) for \( i = 1, 2, \ldots, r \), then we can prove relations between (P) and (D) (see §3) to justify referring to (D) as the dual of (P) and vice versa.

A \( p^{th}\)-order cone programming (POCP) problem in *primal standard form* is

\[
\min \ c^T x \\
\text{s.t.} \ A x = b \\
x \succeq_{r(p)} 0,
\]

where \( m, n, n_1, n_2, \ldots, n_r \) are positive integers such that \( n = \sum_{i=1}^r n_i, \ p \in [1, \infty], \ A \in \mathbb{R}^{m \times n}, \ b \in \mathbb{R}^m \) and \( c \in \mathbb{R}^n \) constitute given data, \( x \in \mathbb{R}^n \) is the primal variable. According to (D), the *dual* problem associated with POCP (6.1.1) is
\[
\begin{align*}
\max & \quad b^T y \\
\text{s.t.} & \quad A^T y + z = c \\
& \quad z \succeq_{r(q)} 0,
\end{align*}
\]

where \( y \in \mathbb{R}^m \) and \( z \in \mathbb{R}^n \) are the dual variables and \( q \) is conjugate to \( p \).

Clearly, second-order cone programs are a special case of POCPs which occurs when \( p = 2 \) in (6.1.1) (hence \( q = 2 \) in (6.1.2)).

**Example 21. Norm minimization problems:**

In [1], Alizadeh and Goldfarb presented second-order cone programming formulations of three norm minimization problems where the norm is the Euclidean norm. In this subsection we indicate extensions of these three problems where we use arbitrary \( p \) norms leading to MOCPs. Let \( v_i = A_i x + b_i \in \mathbb{R}^{n_i-1}, i = 1, 2, \ldots, r \). The following norm minimization problems can be cast as MOCPs:

1. **Minimization of the sum of norms:** The problem \( \min \sum_{i=1}^r \|v_i\|_{p_i} \) can be formulated as

\[
\begin{align*}
\min & \quad \sum_{i=1}^r t_i \\
\text{s.t.} & \quad A_i x + b_i = v_i, \quad i = 1, 2, \ldots, r \\
& \quad (t_1; v_1; t_2; v_2; \ldots; t_r; v_r) \succeq_{(p_1, p_2, \ldots, p_r)} 0.
\end{align*}
\]

2. **Minimization of the maximum of norms:** The problem \( \min \max_{1 \leq i \leq r} \|v_i\|_{p_i} \) can be expressed as the MOCP problem

\[
\begin{align*}
\min & \quad t \\
\text{s.t.} & \quad A_i x + b_i = v_i, \quad i = 1, 2, \ldots, r \\
& \quad (t; v_1; t; v_2; \ldots; t; v_r) \succeq_{(p_1, p_2, \ldots, p_r)} 0.
\end{align*}
\]

3. **Minimization of the sum of the \( k \) largest norms:** More generally, the problem of minimizing the sum of the \( k \) largest norms can also be cast as an MOCP. Let the
norms $\|v_1\|_{p_1}, \|v_2\|_{p_2}, \ldots, \|v_r\|_{p_r}$ be the norms $\|v_1\|_{p_1}, \|v_2\|_{p_2}, \ldots, \|v_r\|_{p_r}$ sorted in nonincreasing order. Then the problem $\min \sum_{i=1}^{r} \|v_i\|_{p_i}$ can be formulated (see also [1] and [23] and the related references contained therein) as the MOCP problem

$$\min \sum_{i=1}^{r} s_i + kt$$

s.t. $A_i x + b_i = v_i$, $i = 1, 2, \ldots, r$

$$(s_1 + t; v_1; s_2 + t; v_2; \ldots; s_r + t; v_r) \succeq \langle p_1, p_2, \ldots, p_r \rangle 0$$

$s_i \geq 0$, $i = 1, 2, \ldots, r$.

### 6.2 Duality

Since MOCPs are a class of convex optimization problems, we can develop a duality theory for them. Here we indicate weak and strong duality for the pair $(P, D)$ as justification for referring to them as a primal dual pair.

It was shown by Faraut and Korányi in [18, Chapter I.2] that the second-order cone is self-dual. We now prove the more general result that the dual of the $p^{th}$-order cone of dimension $n$ is the $q^{th}$-order cone of dimension $n$, where $q$ is the conjugate to $p$.

**Lemma 6.2.1.** $Q_p^* = Q_q$, where $1 \leq p \leq \infty$ and $q$ is the conjugate to $p$. More generally, $Q_{(p_1, p_2, \ldots, p_r)}^* = Q_{(q_1, q_2, \ldots, q_r)}$, where $1 \leq p_i \leq \infty$ and $q_i$ is the conjugate to $p_i$ for $i = 1, 2, \ldots, r$.

**Proof.** The proof of the second part trivially follows from the first part and the fact that $(K_1 \times K_2 \times \cdots \times K_r)^* = K_1^* \times K_2^* \times \cdots \times K_r^*$. To prove the first part, we first prove that $Q_q \subseteq Q_p^*$. Let $x = (x_0; \bar{x}) \in Q_q$, we show that $x \in Q_p^*$ by verifying that $x^T \bar{y} \geq 0$ for any $y \in Q_p$. So let $y = (y_0; \bar{y}) \in Q_p$. Then $x^T y = x_0 y_0 + \bar{x}^T \bar{y} \geq \|x\|_q \|\bar{y}\|_p + \bar{x}^T \bar{y} \geq |\bar{x}^T \bar{y}| + \bar{x}^T \bar{y} \geq 0$, where the first inequality follows from the fact that $x \in Q_q$ and $y \in Q_p$ and the second one from H"older's inequality. Now we show $Q_p^* \subseteq Q_q$.

Let $y = (y_0; \bar{y}) \in Q_p^*$, we show that $y \in Q_q$ by verifying that $y_0 \geq \|\bar{y}\|_q$. This is trivial if
\( \bar{y} = 0 \) or \( p = \infty \). If \( \bar{y} \neq 0 \) and \( 1 \leq p < \infty \), let \( u := (y_1^{p/q}; y_2^{p/q}; \ldots; y_{n-1}^{p/q}) \) and consider \( x := (\|u\|_p; -u) \in Q_p \). Then by using Hölder’s inequality, where the equality is attained, we obtain

\[
0 \leq x^T \bar{y} = \|u\|_p y_0 - u^T \bar{y} = \|u\|_p y_0 - \|u\|_p \|\bar{y}\|_q = \|u\|_p (y_0 - \|\bar{y}\|_q). \]

This gives that \( y_0 \geq \|\bar{y}\|_q \).

From this lemma we deduce that the \( p^{th} \)-order cone is reflexive, i.e., \( Q_p^{**} = Q_p \), and more generally, \( Q_{(p_1, p_2, \ldots, p_r)} \) is also reflexive. On the basis of this fact, it is natural to infer that the dual of the dual is the primal.

In view of the above lemma, problem (D) can be derived from (P) through the usual Lagrangian approach. The Lagrangian function for (P) is

\[
\mathcal{L}(x, \lambda, \nu) = c^T x - \lambda^T (A x - b) - \nu^T x.
\]

The dual objective is

\[
q(\lambda, \nu) := \inf_x \mathcal{L}(x, \lambda, \nu) = \inf_x (c - A^T \lambda - \nu)^T x + \lambda^T b.
\]

In fact, we may call the constraint \( x \succeq_{(p_1, p_2, \ldots, p_r)} 0 \) as the “nonnegativity of \( x \)”, but, with respect to the multi-order cone \( Q_{(p_1, p_2, \ldots, p_r)} \). Note that the Lagrange multiplier \( \nu \) corresponding to the inequality constraint \( x \succeq_{(p_1, p_2, \ldots, p_r)} 0 \) is restricted to be nonnegative with respect to the dual of \( Q_{(p_1, p_2, \ldots, p_r)} \) (i.e., \( \nu \succeq_{(q_1, q_2, \ldots, q_r)} 0 \)), whereas the Lagrange multiplier \( \lambda \) corresponding to the equality constraint \( A x - b = 0 \) is unrestricted. Hence the dual problem is obtained by maximizing \( q(\lambda, \nu) \) subject to \( \nu \succeq_{(q_1, q_2, \ldots, q_r)} 0 \).

If \( c - A^T \lambda - \nu \neq 0 \), the infimum is clearly \(-\infty\). So we can exclude \( \lambda \) for which \( c - A^T \lambda - \nu \neq 0 \). When \( c - A^T \lambda - \nu = 0 \), the dual objective function is simply \( \lambda^T b \).
Table 6.1: Correspondence rules between primal and dual MOCPs.

Hence, we can write the dual problem as follows:

$$\begin{align*}
\max & \quad b^T \lambda \\
\text{s.t.} & \quad A^T \lambda + \nu = c \\
& \quad \nu \succeq_{(q_1,q_2,\ldots,q_r)} 0.
\end{align*}$$

Replacing $\lambda$ by $x$ and $\nu$ by $z$ in (6.2.1) we get (D).

In general, MOCPs can be written in a variety of forms different from the standard form (P, D). The situation in MOCPs is similar to that for linear programs; any MOCP problem can be written in the standard form. However, if we consider MOCPs in other forms, then it is more convenient to apply the duality rules directly. Table 6.1 is a summary of these rules. This table is a generalization of a similar table in [14, Section 4.2].

For instance, using this table, the dual of (P) is the problem

$$\begin{align*}
\max & \quad b^T \lambda \\
\text{s.t.} & \quad A^T \lambda \preceq_{(q_1,q_2,\ldots,q_r)} c,
\end{align*}$$

which is equivalent to problem (6.2.1) where $A^T \lambda \preceq_{(q_1,q_2,\ldots,q_r)} c$ means that $c - A^T \lambda \succeq_{(q_1,q_2,\ldots,q_r)} 0$. 

136
Using Lemma 6.2.1, we can prove the following weak duality property.

**Theorem 6.2.1. (Weak duality)** If $x$ is any primal feasible solution of (P) and $(y, z)$ is any feasible solution of (D), then the duality gap $c^T x - b^T y = x^T z \geq 0$.

**Proof.** Note that $c^T x - b^T y = (A^T y + z)^T x - b^T y = y^T Ax + z^T x - y^T b = y^T (Ax - b) + z^T x = x^T z$. Since $x \in Q_{(p_1, p_2, \ldots, p_r)}$ and $z \in Q_{(q_1, q_2, \ldots, q_r)} = Q_{(p_1, p_2, \ldots, p_r)}^*$, we conclude that $x^T z \geq 0$. □

We now give conditions for strong duality to hold. We say that problem (P) is *strictly* feasible if there exists a primal feasible point $\hat{x}$ such that $\hat{x} \succ_{(p_1, p_2, \ldots, p_r)} 0$. In the remaining part of this section, we assume that the $m$ rows of the matrix $A$ are linearly independent. Using the Karush-Kuhn-Tucker (KKT) conditions, we state and prove the following strong duality result.

**Theorem 6.2.2. (Strong duality I)** Consider the primal–dual pair (P, D). If (P) is strictly feasible and solvable with a solution $x$, then (D) is solvable and the optimal values of (P) and (D) are equal.

**Proof.** By the assumptions of the theorem, $x$ is an optimal solution of (P) where we can apply the KKT conditions. This implies that there are Lagrange multiplier vectors $\lambda$ and $\nu$ such that $(x, \lambda, \nu)$ satisfies

$$
A x = b, \quad x \succeq_{(p_1, p_2, \ldots, p_r)} 0,
A^T \lambda + \nu = c, \quad \nu \succeq_{(q_1, q_2, \ldots, q_r)} 0,
x_i^T \nu_i = 0, \quad \text{for } i = 1, 2, \ldots, r.
$$

This implies that $(\lambda, \nu)$ is feasible for the dual problem (D). Let $(y, z)$ be any feasible solution of (D), then we have that $b^T y \leq c^T x = x^T \nu + b^T \lambda = b^T \lambda$, where we used the weak duality to obtain the inequality and the complementary slackness to obtain the last equality. Thus, $(\lambda, \nu)$ is an optimal solution of (D) and $c^T x = b^T \lambda$ as desired. □
Note that the result in Theorem 6.2.1 is symmetric between (P) and (D). The following strong duality result can also be obtained by applying the duality relations [30, Theorem 4.2.1] to our problem formulation,

**Theorem 6.2.3. (Strong duality II)** Consider the primal–dual pair (P, D). If both (P) and (D) have strictly feasible solutions, then they both have optimal solutions \( x^* \) and \( (y^*, z^*) \), respectively, and \( p^* := c^T x^* = d^* := b^T y^* \) (i.e., \( x^*^T z^* = 0 \) (complementary slackness)).

From the above results, we get the following corollary.

**Corollary 6.2.1. (Optimality conditions)** Assume that both (P) and (D) are strictly feasible, then \( (x, y, z) \in \mathbb{R}^{n+m+n} \) is a pair of optimal solutions if and only if

\[
\begin{align*}
A x &= b, \\
A^T y + z &= c, \\
x_i^T z_i &= 0, \text{ for } i = 1, 2, \ldots, r.
\end{align*}
\]

### 6.3 Multi-order cone programming problems over integers

In this section we introduce two important related problems that result when decision variables in an MOCP can only take integer values. Consider the MOCP problem (P). If we require an additional constraint that a subset of the variables have to attain 0-1 values, then we are interested in optimization problem of the form
\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax = b \\
& \quad x \succeq (p_1, p_2, \ldots, p_r) 0 \\
& \quad x_k \in \{0, 1\}, k \in \Gamma,
\end{align*}
\]

where $\Gamma \subseteq \{1, 2, \ldots, n\}$, the decision variable $x \in \mathbb{R}^n$ has some of its components $x_k$ ($k \in \Gamma$) with integer values and bounded by $\alpha_k, \beta_k \in \mathbb{R}$. This class of optimization problems may be termed as $0$-$1$ multi-order cone programs (0-1MOCPs).

A more general and interesting problem when in a MOCP some variables can only take integer values. If we are given the same data $A, b,$ and $c$ as in (P), then we are interested in the problem of the form

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax = b \\
& \quad x \succeq (p_1, p_2, \ldots, p_r) 0 \\
& \quad x_k \in [\alpha_k, \beta_k] \cap \mathbb{Z}, k \in \Gamma,
\end{align*}
\]

where $\Gamma \subseteq \{1, 2, \ldots, n\}$, the decision variable $x \in \mathbb{R}^n$ has some of its components $x_k$ ($k \in \Gamma$) with integer values and bounded by $\alpha_k, \beta_k \in \mathbb{R}$. This class of optimization problems may be termed as mixed integer multi-order cone programs (MIMOCPs).

### 6.4 Multi-order cone programming problems under uncertainty

In this section we define two-stage stochastic multi-order cone programs (SMCOPs) with recourse to handle uncertainty in data defining (deterministic) MOCPs. Let $r_1, r_2 \geq 1$ be integers. For $i = 1, 2, \ldots, r_1$ and $j = 1, 2, \ldots, r_2$, let $p_{1i}, p_{2j} \in [1, \infty]$ and $m_1, m_2, n_1, n_2, n_{1i},$
be positive integers such that \( n_1 = \sum_{i=1}^{r_1} n_{1i} \) and \( n_2 = \sum_{i=1}^{r_2} n_{2j} \). An SMOCP with recourse in primal standard form is defined based on deterministic data \( A \in \mathbb{R}^{m_1 \times n_1} \), \( b \in \mathbb{R}^{m_1} \) and \( c \in \mathbb{R}^{n_1} \) and random data \( T \in \mathbb{R}^{m_2 \times n_1} \), \( W \in \mathbb{R}^{m_2 \times n_2} \), \( h \in \mathbb{R}^{m_2} \) and \( d \in \mathbb{R}^{n_2} \) whose realizations depend on an underlying outcome \( \omega \) in an event space \( \Omega \) with a known probability function \( P \). Given this data, an SMOCP with recourse in *primal standard form* is

\[
\begin{align*}
\min & \quad c^T x + \mathbb{E}[Q(x, \omega)] \\
\text{s.t.} & \quad Ax = b \\
& \quad x \succeq_{(p_{11}, p_{12}, \ldots, p_{1r_1})} 0,
\end{align*}
\]

where \( x \in \mathbb{R}^{n_1} \) is the first-stage decision variable and \( Q(x, \omega) \) is the minimum value of the problem

\[
\begin{align*}
\min & \quad d(\omega)^T y \\
\text{s.t.} & \quad W(\omega)y = h(\omega) - T(w)x \\
& \quad y \succeq_{(p_{21}, p_{22}, \ldots, p_{2r_2})} 0,
\end{align*}
\]

where \( y \in \mathbb{R}^{n_2} \) is the second-stage variable, and

\[
\mathbb{E}[Q(x, \omega)] := \int_{\Omega} Q(x, \omega) P(d\omega).
\]

Two-stage stochastic \( p^{th} \) \((\text{second})\)-order cone programs with recourse are a special case of SMOCPs with \( p_{1i} = p_{2j} = p \geq 1 \) \((p_{1i} = p_{2j} = 2)\) for all \( i = 1, 2, \ldots, r_1 \) and \( j = 1, 2, \ldots, r_2 \).

### 6.5 An application

Our application is four versions the FLP (see Subsection 5.1.1). For these four versions we present problem descriptions leading to a MOCP model, a 0-1MOCP model, an MIMOCPP
model, and an SMOCP model.

As we mentioned in Subsection 5.1.1, FLPs can be classified based on the distance measure used in the model between the facilities. If we use the Euclidean distance then these problems are called Euclidean facility location problems (EFLPs), if we use the rectilinear distance (also known as $L_1$ distance, city block distance, or Manhattan distance) then these problems are called rectilinear facility location problems (RFLPs). Furthermore, in some applications we use both the Euclidean and the rectilinear distances (based on the relationships between the pairs of facilities) as the distance measures used in the model between the facilities to get a mixed of EFLPs and RFLPs that we refer to as Euclidean-rectilinear facility location problems (ERFLPs). Another way of classification this problem is based on where we can place the new facilities in the solution space. When the new facilities can be placed any place in solution space, the problem is called a continuous facility location problem (CFLP), but usually the decision maker needs the new facilities to be placed at specific locations (called nodes) and not in any place in the solution space. In this case the problem is called a discrete facility location problem (DFLP).

Each one of the next subsections is devoted to a version of ERFLPs. More specifically, we consider (deterministic) continuous Euclidean-rectilinear facility location problems (CERFLPs) which leads to an MOCP model, discrete Euclidean-rectilinear facility location problems (DERFLPs) which leads to a 0-1MOCP model, ERFLPs with integrality constraints which leads to an MIMOCP model, and stochastic continuous Euclidean-rectilinear facility location problems (stochastic CERFLPs) which leads to an SMOCP model.
6.5.1 CERFLPs—An MOCP model

In single ERFLPs, we are interested in choosing a location to build a new facility among existing facilities so that this location minimizes the sum of weighted (either Euclidean or rectilinear) distances to all existing facilities.

Assume that we are given $r + s$ existing facilities represented by the fixed points $a_1, a_2, \ldots, a_r, a_{r+1}, a_{r+2}, \ldots, a_{r+s}$ in $\mathbb{R}^n$, and we plan to place a new facility represented by $x$ so that we minimize the weighted sum of the Euclidean distances between $x$ and each of the points $a_1, a_2, \ldots, a_r$, and the weighted sum of the rectilinear distances between $x$ and each of the points $a_{r+1}, a_{r+2}, \ldots, a_{r+s}$. This leads us to the problem

$$\min \sum_{i=1}^{r} w_i \|x - a_i\|_2 + \sum_{i=r+1}^{r+s} w_i \|x - a_i\|_1$$

or, alternatively, to the problem

$$\min \sum_{i=1}^{r+s} w_i t_i$$

s.t.  
$$(t_1; x - a_1; \ldots; t_r; x - a_r) \succeq_{r(2)} 0$$
$$(t_{r+1}; x - a_{r+1}; \ldots; t_{r+s}; x - a_{r+s}) \succeq_{s(1)} 0,$$

where $w_i$ is the weight associated with the $i$th existing facility and the new facility for $i = 1, 2, \ldots, r + s$.

In multiple ERFLPs we add $m$ new facilities, namely $x_1, x_2, \ldots, x_m \in \mathbb{R}^n$, instead of adding only one. We have two cases depending whether or not there is an interaction among the new facilities in the underlying model. If there is no interaction between the new facilities, we are just concerned in minimizing the weighted sums of the distance between each one of the new facilities and each one of the fixed facilities. In other words, we solve the following MOCP model:
\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r+s} w_{ij} t_{ij} \\
\text{s.t.} & \quad (t_{ij}; x_j - a_1; \ldots; t_{rj}; x_j - a_r) \succeq_{r(2)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (t_{(r+1)j}; x_j - a_{r+1}; \ldots; t_{(r+s)j}; x_j - a_{r+s}) \succeq_{s(1)} 0, \quad j = 1, 2, \ldots, m,
\end{align*}
\]

where \( w_{ij} \) is the weight associated with the \( i \)th existing facility and the \( j \)th new facility for \( j = 1, 2, \ldots, m \) and \( i = 1, 2, \ldots, r + s \).

If interaction exists among the new facilities, then, in addition to the above requirements, we need to minimize the sum of the (either Euclidean or rectilinear) distances between each pair of the new facilities. Let \( 1 \leq l \leq m \) and assume that we are required to minimize the weighted sum of the Euclidean distances between each pair of the new facilities \( x_1, x_2, \ldots, x_l \) and the weighted sum of the rectilinear distances between each pair of the new facilities \( x_{l+1}, x_{l+2}, \ldots, x_m \). In this case, we are interested in a model of the form:

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r+s} w_{ij} t_{ij} + \sum_{j=2}^{m} \sum_{j' = 1}^{j-1} \hat{w}_{jj'} \hat{t}_{jj'} \\
\text{s.t.} & \quad (t_{ij}; x_j - a_1; \ldots; t_{rj}; x_j - a_r) \succeq_{r(2)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (t_{(r+1)j}; x_j - a_{r+1}; \ldots; t_{(r+s)j}; x_j - a_{r+s}) \succeq_{s(1)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\hat{t}_{jj}; x_j - x_{j+1}; \ldots; \hat{t}_{jl}; x_j - x_l) \succeq_{(l-j)(2)} 0, \quad j = 1, 2, \ldots, l-1 \\
& \quad (\hat{t}_{jj}; x_j - x_{j+1}; \ldots; \hat{t}_{jm}; x_j - x_m) \succeq_{(m-j)(1)} 0, \quad j = l + 1, 2, \ldots, m - 1,
\end{align*}
\]

where \( \hat{w}_{jj'} \) is the weight associated with the new facilities \( j' \) and \( j \) for \( j' = 1, 2, \ldots, j - 1 \) and \( j = 2, 3, \ldots, m \).

### 6.5.2 DERFLPs—A 0-1MOCP model

We consider the discrete version of the problem by assuming that the new facilities \( x_1, x_2, \ldots, x_m \) need to be placed at specific locations and not in any place in 2- or 3-
(or higher) dimensional space. Let the points \( \mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k \in \mathbb{R}^n \) represent these specific locations where \( k \geq m \). So, we add the constraint \( \mathbf{x}_i \in \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\} \) for \( i = 1, 2, \ldots, m \).

Clearly, for \( i = 1, 2, \ldots, m \), the above constraint can be replaced by the following linear and binary constraints:

\[
\mathbf{x}_i = \mathbf{v}_1 y_{i1} + \mathbf{v}_2 y_{i2} + \cdots + \mathbf{v}_k y_{ik},
\]

\[
y_{i1} + y_{i2} + \cdots + y_{ik} = 1, \text{ and}
\]

\[
\mathbf{y}_i = (y_{i1}; y_{i2}; \ldots; y_{ik}) \in \{0, 1\}^k.
\]

We also assume that we cannot place more than one facility at each location. Consequently, we add the following constraints:

\[
(1; y_{l1}; y_{l2}; \ldots; y_{lm}) \geq_{(1)} \mathbf{0}, \quad \text{for } l = 1, 2, \ldots, k.
\]

If there is no interaction between the new facilities, then the MOCP model (6.5.1) becomes the following 0-1MOCP model:

\[
\min \quad \sum_{j=1}^{m} \sum_{i=1}^{r+s} w_{ij} t_{ij}
\]

s.t. \( (t_{ij}; \mathbf{x}_j - \mathbf{a}_1; \ldots; t_{rj}; \mathbf{x}_j - \mathbf{a}_r) \geq_{r(2)} \mathbf{0}, \quad j = 1, 2, \ldots, m \)

\( (t_{(r+1)j}; \mathbf{x}_j - \mathbf{a}_{r+1}; \ldots; t_{(r+s)j}; \mathbf{x}_j - \mathbf{a}_{r+s}) \geq_{s(1)} \mathbf{0}, \quad j = 1, 2, \ldots, m \)

\( \mathbf{x}_i = \mathbf{v}_1 y_{i1} + \mathbf{v}_2 y_{i2} + \cdots + \mathbf{v}_k y_{ik}, \quad i = 1, 2, \ldots, m \)

\( (1; y_{l1}; y_{l2}; \ldots; y_{ml}) \geq_{(1)} \mathbf{0}, \quad \text{for } l = 1, 2, \ldots, k \)

\( \mathbf{1}^T \mathbf{y}_i = 1, \quad \mathbf{y}_i \in \{0, 1\}^k, \quad i = 1, 2, \ldots, m. \)

If interaction exists among the new facilities, then the MOCP model (6.5.2) becomes the
following 0-1MOCP model:

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r+s} w_{ij} t_{ij} + \sum_{j=2}^{m} \sum_{j'=1}^{j-1} \hat{w}_{jj'} t_{jj'} \\
\text{s.t.} & \quad (t_{1j}; x_j - a_1; \ldots; t_{rj}; x_j - a_r) \succeq_{r(2)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (t_{(r+1)j}; x_j - a_{r+1}; \ldots; t_{(r+s)j}; x_j - a_{r+s}) \succeq_{s(1)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\hat{t}_{j(j+1)}; x_j - x_{j+1}; \ldots; \hat{t}_{jj}; x_j - x_l) \succeq_{(l-j)(2)} 0, \quad j = 1, 2, \ldots, l-1 \\
& \quad (\hat{t}_{j(j+1)}; x_j - x_{j+1}; \ldots; \hat{t}_{jm}; x_j - x_m) \succeq_{(m-j)(1)} 0, \quad j = l + 1, 2, \ldots, m - 1 \\
& \quad x_i = v_1 y_{i1} + v_2 y_{i2} + \cdots + v_k y_{ik}, \quad i = 1, 2, \ldots, m \\
& \quad 1^T y_i = 1, \quad y_i \in \{0, 1\}^k, \quad i = 1, 2, \ldots, m.
\end{align*}
\]

For \(l = 1, 2, \ldots, k\), let \(z_l = 1\) if the location \(v_i\) is chosen, and 0 otherwise. Then, we can go further, and consider more assumptions: Let \(k_1, k_2, k_3, k_4 \in [1, k]\) be integers such that \(k_1 \leq k_2\) and \(k_3 \leq k_4\). If we must choose at most \(k_1\) of the locations \(v_1, v_2, \ldots, v_{k_2}\), then we impose the constraints:

\[
(k_1; z_1; z_2; \ldots; z_{k_2}) \succeq_{(1)} 0, \quad \text{and} \quad z \in \{0, 1\}^k.
\]

If we must choose at most \(k_1\) of the locations \(v_1, v_2, \ldots, v_{k_2}\), or at most \(k_3\) of the locations \(v_1, v_2, \ldots, v_{k_4}\), then we impose the constraints:

\[
(k_1 f; z_1; z_2; \ldots; z_{k_2}) \succeq_{(1)} 0, \quad (k_3(1-f); z_1; z_2; \ldots; z_{k_4}) \succeq_{(1)} 0, \quad z \in \{0, 1\}^k, \quad \text{and} \quad f \in \{0, 1\}.
\]

### 6.5.3 ERFLPs with integrality constraints—An MIMOCP model

In some problems we may need the locations to have integer-valued coordinates. In most cities, streets are laid out on a grid, so that city is subdivided into small numbered blocks that are square or rectangular. In this case, usually the decision maker needs the new
facility to be placed at the corners of the city blocks. Thus, for each \( i \in \Delta \subset \{1, 2, \ldots, m\} \), let us assume that the variable \( x_i \) lies in the hyperrectangle \( \Xi_i^n \equiv \{x_i : \zeta_i \leq x_i \leq \eta_i, \zeta_i \in \mathbb{R}^n, \eta_i \in \mathbb{R}^n\} \) and has to be integer-valued, i.e. \( x_i \) must be in the grid \( \Xi_i^n \cap \mathbb{Z}^n \). Thus, if there is no interaction between the new facilities, then instead of solving the MOCP model (6.5.1), we solve the following MIMOCP model:

\[
\min \sum_{j=1}^m \sum_{i=1}^{r+s} w_{ij} t_{ij} + \sum_{j=2}^m \sum_{j'=1}^{j-1} \hat{w}_{jj'} \hat{t}_{jj'} \\
\text{s.t.} \quad (t_{1j}; x_j - a_1; \ldots; t_{rj}; x_j - a_r) \succeq_{r(2)} 0, \quad j = 1, 2, \ldots, m \\
(t_{(r+1)j}; x_j - a_{r+1}; \ldots; t_{(r+s)j}; x_j - a_{r+s}) \succeq_{s(1)} 0, \quad j = 1, 2, \ldots, m \\
x_k \in \Xi_k^n \cap \mathbb{Z}^n, \quad k \in \Delta.
\]

If interaction exists among the new facilities, then instead of solving the MOCP model (6.5.2), we solve the following MIMOCP model:

\[
\min \sum_{j=1}^m \sum_{i=1}^{r+s} w_{ij} t_{ij} + \sum_{j=2}^m \sum_{j'=1}^{j-1} \hat{w}_{jj'} \hat{t}_{jj'} \\
\text{s.t.} \quad (t_{1j}; x_j - a_1; \ldots; t_{rj}; x_j - a_r) \succeq_{r(2)} 0, \quad j = 1, 2, \ldots, m \\
(t_{(r+1)j}; x_j - a_{r+1}; \ldots; t_{(r+s)j}; x_j - a_{r+s}) \succeq_{s(1)} 0, \quad j = 1, 2, \ldots, m \\
(t_{(l+1)j}; x_j - x_{j+1}; \ldots; \hat{t}_{jl}; x_j - x_l) \succeq_{(l-j)(2)} 0, \quad j = 1, 2, \ldots, l-1 \\
(t_{(l+1)j}; x_j - x_{j+1}; \ldots; \hat{t}_{jm}; x_j - x_m) \succeq_{(m-j)(1)} 0, \quad j = l + 1, 2, \ldots, m-1 \\
x_k \in \Xi_k^n \cap \mathbb{Z}^n, \quad k \in \Delta.
\]

6.5.4 Stochastic CERFLPs—An SMOCP model

Before we describe the stochastic version of this generic application, we indicate a more concrete version of it. Assume that we have a new growing city with many suburbs and we want to build a hospital for treating the residents of this city. Some people live in the city at the present time. As the city expands, many houses in new suburbs need to be built and the locations of these suburbs will be known in the future in different parts of the city. Our goal is to find the best location of this hospital so that it can serve the
current suburbs and the new ones. This location must be determined at the current time and before information about the locations of the new suburbs become available. For those houses that are close enough to the location of the hospital, we use the rectilinear distance as a distance measure between them and the hospital, while for the new suburbs that will be located far away from the location of the hospital, we use the Euclidean distance as a distance measure between them and the hospital. See Figure 6.1.

**Figure 6.1:** A more concrete version of the stochastic CERFLP: A new growing city with many expected building houses in different possible sides of the city.

Generally speaking, let $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{r_1}, \mathbf{a}_{r_1+1}, \mathbf{a}_{r_1+2}, \ldots, \mathbf{a}_{r_1+s_1}$ be fixed points in $\mathbb{R}^n$ representing the coordinates of $r_1 + s_1$ existing fixed facilities and $\tilde{\mathbf{a}}_1(\omega), \tilde{\mathbf{a}}_2(\omega), \ldots, \tilde{\mathbf{a}}_{r_2}(\omega), \tilde{\mathbf{a}}_{r_2+1}(\omega), \tilde{\mathbf{a}}_{r_2+2}(\omega), \ldots, \tilde{\mathbf{a}}_{r_2+s_2}(\omega)$ be random points in $\mathbb{R}^n$ representing the coordinates of $r_2 + s_2$ random facilities whose realizations depends on an underlying outcome $\omega$ in an event space $\Omega$ with a known probability function $P$.

Suppose that at present we do not know the realizations of $r_2 + s_2$ random facilities, and that at some point in time in future the realizations of these $r_2 + s_2$ random facilities
become known.

Our goal is to locate \( m \) new facilities \( x_1, x_2, \ldots, x_m \in \mathbb{R}^n \) that minimize the following sums:

- the weighted sums of the Euclidean distance between each one of the new facilities and each one of the fixed facilities \( a_1, a_2, \ldots, a_{r_1}; \)
- the weighted sums of the rectilinear distance between each one of the new facilities and each one of the fixed facilities \( a_{r_1+1}, a_{r_1+2}, \ldots, a_{r_1+s_1}; \)
- the expected weighted sums of the Euclidean distance between each one of the new facilities and each one of the random facilities \( \tilde{a}_1(\omega), \tilde{a}_2(\omega), \ldots, \tilde{a}_{r_2}(\omega); \)
- the expected weighted sums of the rectilinear distance between each one of the new facilities and each one of the random facilities \( \tilde{a}_{r_2+1}(\omega), \tilde{a}_{r_2+2}(\omega), \ldots, \tilde{a}_{r_2+s_2}(\omega). \)

Note that this decision needs to be made before the realizations of the \( r_2 + s_2 \) random facilities become available. If there is no interaction between the new facilities, then the (deterministic) MOCP model (6.5.1) becomes the following SMOCP model:

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_1+s_1} w_{ij} t_{ij} + \mathbb{E} \left[ Q(x_1; \ldots; x_m, \omega) \right] \\
\text{s.t.} & \quad (t_{1j}; x_j - a_1; \ldots; t_{r_1j}; x_j - a_{r_1}) \succeq_{r_1(2)} 0, \; j = 1, 2, \ldots, m \\
& \quad (t_{(r_1+1)j}; x_j - a_{r_1+1}; \ldots; t_{(r_1+s_1)j}; x_j - a_{r_1+s_1}) \succeq_{s_1(1)} 0, \; j = 1, 2, \ldots, m,
\end{align*}
\]

where \( Q(x_1; \ldots; x_m, \omega) \) is the minimum value of the problem

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_2+s_2} \tilde{w}_{ij}(\omega) \tilde{t}_{ij} \\
\text{s.t.} & \quad (\tilde{t}_{1j}; x_j - \tilde{a}_1(\omega); \ldots; \tilde{t}_{r_2j}; x_j - \tilde{a}_{r_2}(\omega)) \succeq_{r_2(2)} 0, \; j = 1, 2, \ldots, m \\
& \quad (\tilde{t}_{(r_2+1)j}; x_j - \tilde{a}_{(r_2+1)}(\omega); \ldots; \tilde{t}_{(r_2+s_2)j}; x_j - \tilde{a}_{(r_2+s_2)}(\omega)) \succeq_{s_2(1)} 0, \; j = 1, 2, \ldots, m,
\end{align*}
\]
and

\[ \mathbb{E}[Q(x_1; \ldots; x_m, \omega)] := \int_{\Omega} Q(x_1; \ldots; x_m, \omega) P(d\omega), \]

where \( w_{ij} \) is the weight associated with the \( i \)th existing facility and the \( j \)th new facility for \( j = 1, 2, \ldots, m \) and \( i = 1, 2, \ldots, r_1 + s_1 \), and \( \hat{w}_{ij}(\omega) \) is the weight associated with the \( i \)th random existing facility and the \( j \)th new facility for \( j = 1, 2, \ldots, m \) and \( i = 1, 2, \ldots, r_2 + s_2 \).

If interaction exists among the new facilities, then the (deterministic) MOCP model (6.5.2) becomes the following SMOCP model:

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_1+s_1} w_{ij} t_{ij} + \sum_{j=1}^{m} \sum_{j'=1}^{j-1} \hat{w}_{jj'} \hat{t}_{jj'} + \mathbb{E} \left[ Q(x_1; \ldots; x_m, \omega) \right] \\
\text{s.t.} & \quad (t_{1j}; x_j - a_1; \ldots; t_{r_1j}; x_j - a_{r_1}) \succeq_{r_1(2)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (t_{(r_1+1)j}; x_j - a_{r_1+1}; \ldots; t_{(r_1+s_1)j}; x_j - a_{r_1+s_1}) \succeq_{s_1(1)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\hat{t}_{(j+1)j}; x_j - x_{j+1}; \ldots; \hat{t}_{jj}; x_j - x_i) \succeq_{(l-j)(2)} 0, \quad j = 1, 2, \ldots, l - 1 \\
& \quad (\hat{t}_{(j+1)j}; x_j - x_{j+1}; \ldots; \hat{t}_{jm}; x_j - x_m) \succeq_{(m-j)(1)} 0, \quad j = l + 1, 2, \ldots, m - 1,
\end{align*}
\]

where \( Q(x_1; \ldots; x_m, \omega) \) is the minimum value of the problem.

\[
\begin{align*}
\min & \quad \sum_{j=1}^{m} \sum_{i=1}^{r_2+s_2} \hat{w}_{ij}(\omega) \hat{t}_{ij} + \sum_{j=1}^{m} \sum_{j'=1}^{j-1} \hat{w}_{jj'} \hat{t}_{jj'} \\
\text{s.t.} & \quad (\hat{t}_{1j}; x_j - \hat{a}_1(\omega); \ldots; \hat{t}_{r_2j}; x_j - \hat{a}_{r_2}(\omega)) \succeq_{r_2(2)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\hat{t}_{(r_2+1)j}; x_j - \hat{a}_{(r_2+1)}(\omega); \ldots; \hat{t}_{(r_2+s_2)j}; x_j - \hat{a}_{(r_2+s_2)}(\omega)) \succeq_{s_2(1)} 0, \quad j = 1, 2, \ldots, m \\
& \quad (\hat{t}_{(j+1)j}; x_j - x_{j+1}; \ldots; \hat{t}_{jj}; x_j - x_i) \succeq_{(l-j)(2)} 0, \quad j = 1, 2, \ldots, l - 1 \\
& \quad (\hat{t}_{(j+1)j}; x_j - x_{j+1}; \ldots; \hat{t}_{jm}; x_j - x_m) \succeq_{(m-j)(1)} 0, \quad j = l + 1, 2, \ldots, m - 1,
\end{align*}
\]

and

\[ \mathbb{E}[Q(x_1; \ldots; x_m, \omega)] := \int_{\Omega} Q(x_1; \ldots; x_m, \omega) P(d\omega), \]

where \( \hat{w}_{jj'} \) is the weight associated with the new facilities \( j' \) and \( j \) for \( j' = 1, 2, \ldots, j - 1 \) and \( j = 2, 3, \ldots, m \).
Chapter 7

Conclusion

In this dissertation we have introduced the two-stage stochastic symmetric programming (SSP) with recourse. While one can take the viewpoint that SSPs are a way of dealing with uncertainty in data defining deterministic symmetric programs, it is also possible to take the dual viewpoint that SSPs are a way of allowing any symmetric cone in place of the nonnegative orthant cone in stochastic linear programs, or the positive semidefinite cone in stochastic semidefinite programs. It is also seen that the SSP problem includes some important general classes of optimization problems (Problems 1-6) as special cases.

We have presented a logarithmic barrier interior point algorithm for solving SSPs using Bender’s decomposition. We have proved the convergence results by showing that the logarithmic barrier associated with the recourse function is a strongly self-concordant barrier on the first-stage solutions. We have described and analyzed short- and long-step variance of the algorithm that follow the primal central trajectory of the first-stage problem. We concluded that, for given two symmetric cones with ranks $r_1$ and $r_2$ in the first- and second-stage problems, respectively, and for $K$ number of realizations, we need at most $O(\sqrt{r_1 + Kr_2} \ln(\mu^0/\epsilon))$ Newton iterations in the short-step class of the algorithm to follow the first-stage central path from a starting value of the barrier $\mu^0$ to a terminating value $\epsilon$. We have also shown that at most $O((r_1 + Kr_2) \ln(\mu^0/\epsilon))$ Newton iterations for
this recentering in the long-step class of the algorithm.

An alternative to the logarithmic barrier is the volumetric barrier of Vaidya [40]. In this dissertation, we have also presented a class of volumetric barrier decomposition algorithms for SSPs. We have proved the convergence results by showing that the volumetric barrier associated with the recourse function is a strongly self-concordant barrier on the first-stage solutions. We have also described and analyzed short- and long-step variant of the algorithm that follow the primal central trajectory of the first-stage problem. For given two symmetric cones with ranks $r_1$ and $r_2$ and dimensions $n_1$ and $n_2$ in the first- and second-stage problems, respectively, and for $K$ number of realizations, we have seen that in order to follow the first-stage central path from a starting value of the barrier $\mu^0$ to a terminating value $\epsilon$, we need at most $O((1 + K)(m_1 c_1 + m_2 c_2) \ln(\mu^0/\epsilon))$ Newton iterations in the short-step class of the algorithm. We have also shown that at most $O((1 + K)(m_1 c_1 + m_2 c_2) \ln(\mu^0/\epsilon))$ Newton iterations in the long-step class of the algorithm.

Chen and Mehrotra [16] have proposed a prototype primal interior point decomposition algorithm for the two-stage stochastic convex optimization problem. Since stochastic symmetric programming is a subclass of general stochastic convex optimization, our problem can be solved by using their prototype algorithm, but doing this is not a good idea, because solving SSPs as convex programs without exploiting the special structure of the symmetric cone is not advisable both theoretically and practically. Therefore, a separate study of SSP algorithms is warranted.

Concerning applications, we have presented four application areas that lead to SSPs when the underlying symmetric cones are second-order cones and rotated quadratic cones. This has been simply accomplished by applying a relaxation on assumptions that include deterministic data to include random data.

Problems 1-6 in Chapter 2 are all important special cases of the SSP problem. So, it is interesting to investigate other applicational settings leading to these general problems, such as the ones obtained when the underlying symmetric cones are real symmetric or...
complex Hermitian positive semidefinite matrices.

Concerning implementation of the proposed SSP algorithms, it will be interesting to implement these methods in the future to determine their practical effects. We therefore refer ourselves to a paper by Mehrotra and Özevin [26] where they develop efficient practical implementations of their SSDP algorithm proposed in [25].

We have ended the dissertation by introducing deterministic and stochastic multi-order cone programs (MOCPs) as new conic optimization problems and to be natural extensions of deterministic and stochastic second-order cone programs. We presented an application leading to deterministic, 0-1, integer, and stochastic multi-order cone programs. The MOCP problems are over non-symmetric cones, hence we left such problems as “logarithmically unsolved” open problems so as to contemplate for our future research in conic programming.
Bibliography


